NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

DEVELOPMENT OF A FLIGHT SIMULATION
CONCEPT AND AERODYNAMIC BUILDUP
FOR INVESTIGATION OF DEPARTURE
PREVENTION SYSTEMS IN TACTICAL AIRCRAFT

by

Albert Lawrence Raithel, III

September 1983

Thesis Advisor: Marle D. Hewett

Approved for public release, distribution unlimited.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Development of a Flight Simulation Concept and Aerodynamic Buildup for Investigation of Departure Prevention Systems in Tactical Aircraft		5. TYPE OF REPORT & PERIOD COVERED Masters Thesis September 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Albert Lawrence Raithel, III		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 87
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

aerodynamic buildup, aerodynamic equations, programming flight simulation, stability and control, handling qualities, flying qualities, computer simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The conceptual development of a computer flight simulation for design, testing and analysis of departure prevention systems, simulation capability and programming are discussed, along with required research material and data. A description is given of the aerodynamic buildup program written for

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Development of a Hybrid Simulation Program for Aerodynamic Testing for Investigation of Departure Prevention Systems in Tactical Aircraft		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; [illegible]
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Albert Lawrence [illegible], II		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE [illegible]
		13. NUMBER OF PAGES [illegible]
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Aerodynamic; [illegible], [illegible] simulation, [illegible] stability and control, [illegible] simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The conceptual development of a [illegible] for design, testing and analysis of departure prevention systems. [illegible] programs are discussed, along with simulated [illegible]. A description is given of the aerodynamic [illegible] program [illegible].

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

incorporation in the simulation, including the aerodynamic equations of the model base aircraft, sample program statements and output.

incorporation in pseudoderivation, including the aerodynamic equations of the model base aircraft, sample program statements and output.

Approved for public release, distribution unlimited.

Development of a Flight Simulation Concept and Aerodynamic Buildup for Investigation of Departure Prevention Systems in Tactical Aircraft

by

Albert Lawrence Raithel, III
Lieutenant, United States Navy
B.S., United States Naval Academy, 1976

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1983

ABSTRACT

The conceptual development of a computer flight simulation for design, testing and analysis of departure prevention systems, simulation capability and programming are discussed, along with required research material and data. A description is given of the aerodynamic buildup program written for incorporation in the simulation, including the aerodynamic equations of the model base aircraft, sample program statements and output.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The author wishes to express his appreciation to the following individuals for their assistance in completing work for this thesis: Dr. Marle D. Hewett, Naval Postgraduate School, Thesis Advisor; Capt. Albert L. Raithel, Jr., USN (Ret), VEDA, Inc.; Mr. John Hodgkinson, Dr. Dinesh Joshi, Mr. Peter Shaw, Mr. Larry Hartsook, Mr. Juri Kalviste and Mr. Rod Davis, Controls Research Division, Northrop Aircraft Corp.; Mr. Al Piranian, Naval Air Development Center; Mr. Lee Duke, NASA Dryden Flight Research Center; Mr. Tom Lawrence, Mr. Jim Rebel and Mr. Jonah Ottensoser, Naval Air Systems Command; CDR Wayne Durham, CDR Vern Gordon and LT Dan Kuper, Naval Postgraduate School.

I. INTRODUCTION

Throughout the history of aviation, departure from controlled flight has been a persistent problem. Departure has occurred during various periods of aviation history for different reasons. In the early years, it was an inadequate knowledge of aerodynamic effects leading to poor or inadequate designs. In more recent years, modern design techniques and an improved understanding of aerodynamics, and stability and control have led to the design of high performance aircraft which constantly fly at the limits of their operating envelopes and that in less than a seconds time can be outside of that envelope departing controlled flight. In past times, recovery from departure was often a relatively easy procedure. It still is with simple, basic, fundamental, stable aircraft designs. Recent state-of-the-art tactical aircraft, however, realize their capabilities by displaying neutral or unstable static stability compensated for by digital fly-by-wire control systems. These aircraft with their instabilities and non-conventional aerodynamic design features are not so easily recovered.

As aircraft control systems have been developed over the years, many and varied departure control, departure prevention and departure recovery systems have been developed and flown. The majority of these systems have been limiting type systems, which in some way limit the operation of the aircraft; an angle-of-attack limiter being a common example.

During the performance of an aircraft mission, an actual departure, whether controlled or uncontrolled, recoverable or unrecoverable, will

result in at least the loss of mission effectiveness and probably the loss of man or aircraft or both. By the same means, restructing aircraft operation to levels below the maximum designed capability in order to avoid potential departure situations may result in the same losses of mission, man and/or aircraft. For these reasons it is desirable to develop a departure prevention system for tactical aircraft that is as "non-limiting" as possible.

This thesis is the first report on the development of a computer flight simulation for the design, testing and analysis of modern optimal, adaptive departure systems. It contains the results of project definition and planning, and the details of the aerodynamic buildup developed for incorporation in the flight simulation program package.

II. SIMULATION CONCEPT DESCRIPTION

A. SIMULATION CAPABILITY

The development of a flight simulation is very dependent on the purpose for which it will be utilized. A full flight simulation is required for full motion base simulator, whereas a much reduced version may be used for investigation of carrier landing characteristics. The following are some of the key points considered and decisions made in determining the type and extent of the simulation needed for this project.

1. Although data indicates that departure is still a problem in older tactical aircraft, the application of modern active control techniques to departure systems is most applicable to fly-by-wire or control-by-wire systems.

2. Availability of data led to utilization of the McDonnell Douglas F/A-18A as the simulation data base.

3. The desire to avoid the additional knowns and unknowns of supersonic flight performance reduced the simulation speed envelope to the subsonic regime.

4. For the most applicable case, the simulation will involve up-and-away flight conditions only.

5. The outer loop closures of the aircraft automatic flight control system will not be simulated but in its place an outer-loop maneuvering autopilot will be modeled. The aircraft control augmentation system will be simulated.

6. Given the above conditions and the potential to depart flight throughout the entire flight envelope the full aircraft system in terms of operating limits, control laws and systems will be modeled as closely as possible to the model base aircraft.

The resulting flight simulation will be comparable with other digital fly-by-wire aircraft. Controlled maneuvers will be precisely performed and repeatable via the maneuvering autopilot and the performance and flying qualities should match closely with that of the F/A-18.

B. SIMULATION PROGRAMMING

The programming of a flight simulation generally consists of three major components, a flight control laws model, an aerodynamic buildup, and flight dynamics calculations. Each of these components is quite complex in itself with the entire simulation requiring several programmers. This results in a modular type programming with each of the three components comprising a module. This is an optimum situation in that each module, control laws, aerodynamic buildup, and flight dynamic performs different calculations for which programming can be specifically tailored. Once programmed, each module can be tested by test stubs to verify results prior to inclusion in the full flight simulation program. The use of modular programming reduces the complexity of the simulation and allows identification of real and potential problems in the simulation by testing each module through the full range of flight conditions. The tailoring of the programming for the various modules led to the utilization of both CSMP and FORTRAN computer languages, in the simulation. The appropriate language is utilized in the simulation where the following characteristics are advantageous;

CSMP:

- The capability to handle nonlinear and time-invariant problems.
- The provisions to allow the modeling/simulation of a physical system utilizing block diagrams.

FORTRAN:

- The capability to handle a large quantity of data.
- The capability for formatted output.
- The capability for logic, branching and subroutines.

CSMP is generally used for the dynamic flight control laws model, the flight dynamics calculations, the program controls and unformatted output. FORTRAN language is used for the aerodynamic buildup, gain functions, other minor functions where necessary and the simulation formatted output.

The use of each language where appropriate results in a faster, more accurate, more efficient flight simulation.

C. SIMULATION FORMAT AND OPERATION

The flight simulation program format consists of the three major modules; flight control laws model, aerodynamic buildup and flight dynamics calculations along with a program explanations section, a program control and output section and minor subroutines and functions. The following is a brief description of the operation of the flight simulations program (see Figure 1). Command inputs are made to the dynamic flight control laws model (CSMP), the output of which are control surface deflections, of leading edge and trailing edge flaps, ailerons, horizontal stabilizer, rudder and speedbrake. These surface deflections are input to the aerodynamic buildup (FORTRAN) the output of which

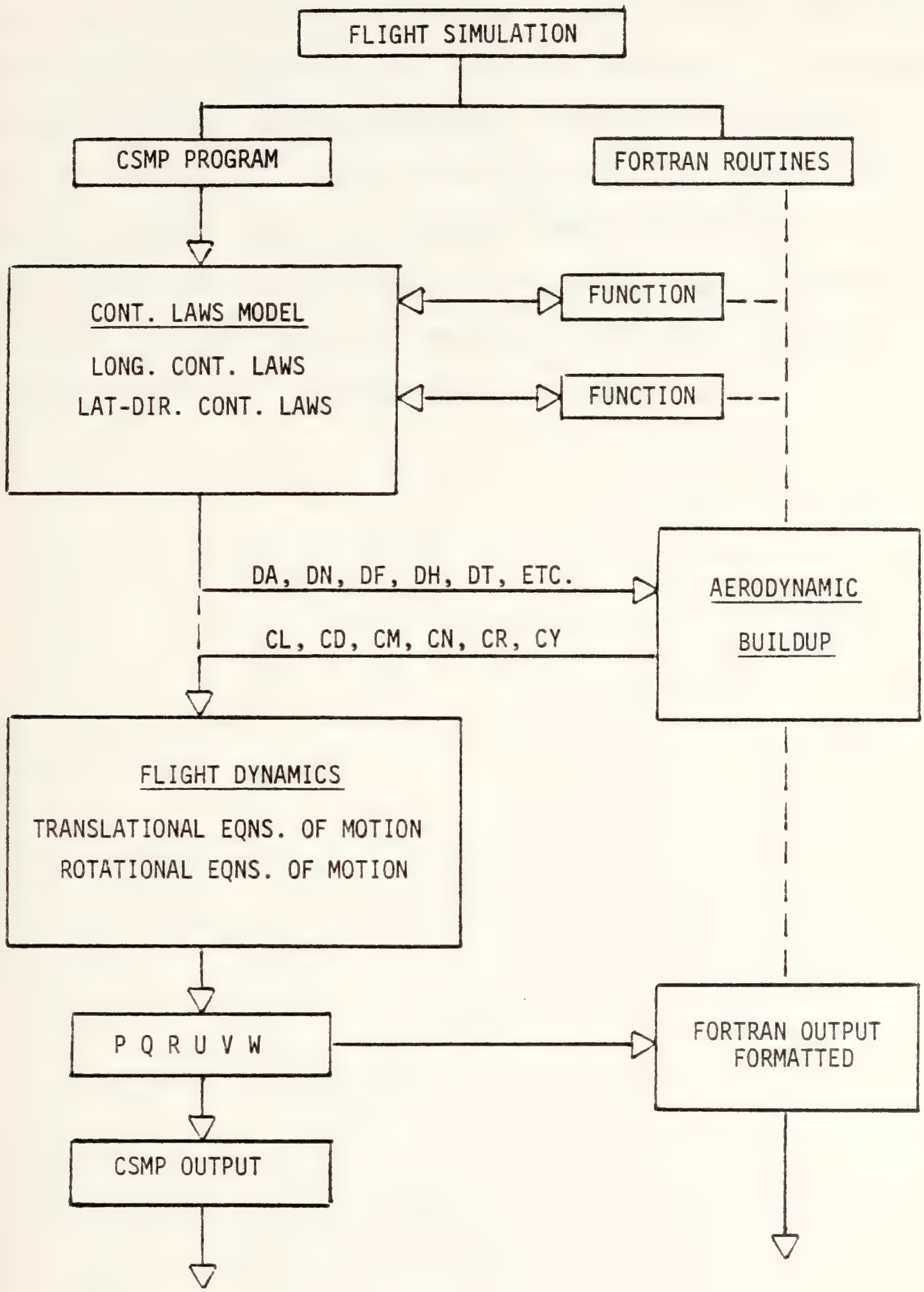
FLIGHT SIMULATION
CSMP PROGRAM
FORTRAN ROUTINES
CONT. LAWS MODEL
LONG. CONT. LAWS
LAT-DIR. CONT. LAWS
FUNCTION
FUNCTION
DA, DN, DF, DH, DT, ETC.
CL, CD, CM, CN, CR, CY
AERODYNAMIC
BUILDUP
FLIGHT DYNAMICS
TRANSLATIONAL EQNS. OF MOTION
ROTATIONAL EQNS. OF MOTION
P Q R U V W
FORTRAN OUTPUT
FORMATTED
CSMP OUTPUT

Figure 1

are the aircraft total coefficients for lift, drag, pitching moment, rolling moment, yawing moment and side force. These coefficients are inputs to the flight dynamics module where aircraft rotational and translational motions of pitch rate, roll rate, yaw rate and U, V, W velocities are computed from the equations of motion. The aircraft motions are fed back to the command input side of the flight control laws module for comparison to commanded inputs and subsequent command modification. The program run time, integration time and other control functions are input from the CSMP program. Output is generated from both CSMP statements and FORTRAN subroutines for formatting.

III. DATA AND RESOURCES

In developing a flight simulation, two types of information are needed: (1) required information - flight systems description, etc., and (2) reference information - programming options, etc. Reviewing wide range of tasks required for a simulation of this magnitude the need to have a source library is obvious. The project has four distinct tasks to be performed, (1) project definition and planning, (2) mathematical modeling, (3) programming and (4) testing and analysis. Research material and required data was collected in each of these areas for use in completing the tasks. The collected material can be divided into six areas, (1) General Departure Information, (2) Aerodynamic Data, (3) Flight Control Laws, (4) Maneuvering Autopilots, (5) Programming Techniques, and (6) Flying Qualities. The following is a list of the major resources obtained for the flight simulation project, and a brief description of each.

A. GENERAL DEPARTURE INFORMATION

Reference 1 contains all mishap reports from mishaps classified by type as uncontrolled flight. It is subdivided into jet, prop and helicopter mishaps and provides information on mishap causes, phase of mission and a narrative of the mishap.

B. FLIGHT CONTROL LAWS

Reference 2 is a description of the inner and outer loop control laws. It is presented in three sections as follows:

1. Flight Control System Characteristics: Inner Loop Theory of Operation. This section contained information on the longitudinal, lateral and directional control laws, quad sensor signals, actuator systems, angle-of attack system and air data system.

2. Automatic Flight Control System: Theory of Operation.

3. Autothrottle: Theory of Operation.

Reference 3 contains system descriptions and diagrams of the following systems pertinent to a flight simulation: longitudinal and lateral-directional control systems, flap commands, mechanical primary controls, flight control electronic set, actuation devices, and throttle control.

Reference 4 is the F/A-18 version 8.2.1 flight control system description and theory of operation. It contains a description of the flight control hardware and interfaces and the system theory of operation including software architecture and mathematical characteristics of inner and outer loop control laws.

C. STABILITY AND CONTROL

Reference 5 contains the stability and control characteristics of the production F/A-18 high speed maneuvering and high lift configurations, derived from wind tunnel test and revised where appropriate to reflect the results of developmental flight tests. The report presents data in a graphical form for status longitudinal and lateral-directional stability and control, and the longitudinal, and lateral-directional dynamic derivatives.

D. FLYING/HANDLING QUALITIES

Reference 6 presents the flying and handling qualities of the F/A-18 figher escort configuration. Longitudinal and lateral-directional modes

and responses, unaugmented characteristics, and spin departure characteristics are included. Information is in both graphical and tabular form.

E. MANEUVERING AUTOPILOT

Reference 7 is a discussion of developing a maneuvering autopilot. It includes maneuvering requirements, linear analysis and design, control law development, command generation and flight experience.

F. PROGRAMMING TECHNIQUES

Information on programming techniques for manipulating large qualities of data with emphasis on flight simulations and aerodynamic buildups was obtained from both Northrop Aircraft Corp. and the Naval Air Development Center.

This is by no means a complete list of the information obtained. It is, however, the primary material used during the project. It is discussed to indicate the type of materials required to develop a flight simulation. The general departure material was used to determine what flight conditions should be investigated. The flight control laws material is being utilized to develop the dynamic flight control law model. The stability and control data is used in the aerodynamic buildup. The maneuvering autopilot data is used for modeling the outer loop maneuvering autopilot. The programming techniques material is used for programming methodology and the flying qualities data is used for verification of simulation model response.

IV. AERODYNAMIC BUILDUP

A. CONSIDERATIONS

As discussed earlier, the major parts to a flight simulation program are a flight control laws model, an aerodynamic buildup and flight dynamics calculations. The following is a description of the aerodynamic buildup developed for incorporation into the flight simulation program. In developing the buildup, the following goals were set.

1. Simplicity and intelligibility.

2. Ability to operate as a separate program or be incorporated as a subprogram in a larger simulation.

3. Provide proper results throughout the entire range of flight conditions for the simulation.

4. Flexibility, versatility and alterability.

The aerodynamic buildup constitutes a large portion of the entire simulation. It also involves the manipulation of very large quantities of data. Its programming must consider, integration with other program modules, data handling times and storage space. These considerations impact on decisions about programming language, programming methodology, and data storage and retrieval techniques.

B. AERODYNAMIC EQUATIONS

The operation of the flight simulation program, discussed in Chapter Two, indicated the inputs to the aerodynamic buildup are the aircraft control surface deflections and the outputs are the aircraft aerodynamic coefficients. The first task was the determination of what control

surface deflections and flight conditions affected each coefficients and to what extent. For example, lift coefficient is changed by deflecting the horizontal stabilizer. How much it is changed is determined by the amount of deflection, the airspeed, and the angle-of-attack. This information was determined from the model base aerodynamic equations [Ref. 4]. Below is a list of the control surfaces and flight conditions affecting each coefficient. The complete aerodynamic equations with definitions and explanations are provided in the appendix.

1. Lift Coefficient is a function of: Mach No., altitude, angle-of-attack, leading-edge flap (LEF) deflection, trailing-edge flap (TEF) deflection, horizontal tail deflection, speedbrake deflection, aileron deflection, pitch rate and angle-of-attack rate.

2. Drag Coefficient is a function of: Mach No., angle-of-attack, LEF deflection, TEF deflection, horizontal tail deflection, aileron deflection and speedbrake deflection.

3. Pitching Moment Coefficient is a function of: same as lift coefficient with the addition of rudder deflection.

4. Yawing Moment Coefficient is a function of: Mach No., altitude, angle-of-attack, sideslip angle, LEF deflection, TEF deflection, differential tail deflection, speedbrake deflection, rudder deflection, aileron deflection, roll rate and yaw rate.

5. Rolling Moment is a function of: same as yawing moment with the addition of flaperon or differential TEF deflection.

6. Side Force Coefficient is a function of: same as yawing moment.

C. AERODYNAMIC DATA

Once the aerodynamic equations were obtained the next task was to obtain the value of each term in each equations for given flight conditions or, the aerodynamic data. This data, presented graphically [Ref. 4] was derived from wind tunnel testing but updated where possible by developmental flight test results. The data was given for low angle-of-attack and high angle-of-attack, considered to be forty degrees or higher. The distinction exists for the following reason: Above forty degrees angle-of-attack the leading-edge flaps are fixed to 34 degrees and the trailing-edge flaps are undeflected. This is the configuration used in measuring the basic coefficients and no increments are added for leading or trailing-edge flaps. Below 40° angle-of-attack the basic coefficients are measured at the zero flap deflections configuration and increments are added for leading and trailing edge flap deflections as necessary.

Data was available for most of the flight envelope. In instances where no data was available, such as high angle-of-attack speedbrake data, the increments were set to zero. If for some increment data was not available throughout the desired ranges, judgment was made to determine the increment in one of three ways. 1) If the data reports noted that linear interpolation was possible, then the value was so obtained, 2) If it appeared that the increment was approaching to be zero, realistically it was made to go to zero or, 3) If no other indications existed, the increment was left constant throught the range. As an example, consider yawing moment increment due to speedbrake deflection. The data was presented for sideslip angles of positive two and ten degrees. The incremental changes were required over a sideslip angle range from negative twenty to positive twenty degrees.

The discrepancy was solved as follows. The increments were lineraly interpolated between zero and ten degrees and then held constant from ten to twenty degrees. The negative sideslip angle values were determined by using the negative of the positive sideslip angle values. These adjustments to the actual aerodynamic data comprise a very small percentage of the data. They do not occur in any ciritical values of flight conditions and are determined realistically enough to have no adverse effect on the validity of the simulation. In contrast by covering the complete range of flight conditions, the aerodynamic buildup provides for a more realistic simulation. Once the aerodynamic data were obtained and evaluated, they had to be extracted from the graphical form to tabular form for computer entry. The values of flight conditions and surface deflections for which data are tabulated is presented in the appendix.

D. PROGRAMMING

As originally envisioned, the aerodynamic buildup would be an integral part of the main simulation CSMP program. This approach quickly ran into problems with handling functions of three and four variables, large quantities of numbers and sorting techniques. It was decided to program the aerodynamic buildup as a FORTRAN routine used as a subprogram in the flight simulation. The additional requirement of providing aircraft coefficients continuously throughout the flight envelope from aerodynamic data tabulated at specific intervals led to the use of a table look-up routine with interpolation functions, for intermediate flight conditions. The program procedure is exactly the same for each coefficient as follows:

1. A data file exists holding all the values of all the terms in the aerodynamic equation for given flight conditions.

2. The program reads the data file and loads the data into program matrices. There then exists a separate data matrix for each term in the respective coefficients aerodynamic equation.

3. These matrices are then printed out to display the data being used in the buildup.

4. The program then makes calls to interpolation subroutines to determine the actual value of each term in the aerodyanmic equation for the existing flight conditions.

5. The terms are then added appropriately to form the total coefficient.

An example of the program for lift coefficient is provided in the appendix. Each coefficient varies only in the terms of the aerodynamic equation.

The aerodynamic buildup program is segmented into six major sections.

Section One: Variable definition, explanations, declarations and program parameters. This section contains FORTRAN declaration and dimension statements, program operation notes and control cards. The control cards provide the options for obtaining or deleting hardcopy output of the aerodynamic data, and the computed derivatives and coefficients. Additionally, the user can select to use test input flight conditions or inputs from another source, such as the main simulation program.

Section Two: Aerodynamic data and constants. This section contains the read and write statements for each of the six coefficients to load the program data matrices and provide hardcopy output of the tabulated data, if desired.

Section Three: Test flight condition inputs. This section provides the operator the input of test flight conditions and control surface conditions. Standard day atmospheric tables incorporated in the program can also be selected if desired. This section can be totally deleted when the program is used as a subprogram to a larger simulation.

Section Four: Aerodynamic buildup. This section contains for each coefficient the interpolation subroutine call statements to the data matrices to determine the value of the terms in the aerodynamic equations. The total coefficients are actually determined in this section by summing the terms in the respective equations.

Section Five: Output. This section contains the format statements for the formatted output of hardcopy data and results.

Section Six: Subroutines. This section contains the interpolation subroutines used by the program. There are four subroutines, one each for functions of one, two, three or four variables.

V. SUMMARY

There are no conclusions to draw for this report. A few comments can be made on the work that was done. The flight simulation project, initial project definition and planning was completed and the project is well underway. The initial concepts and requirements are still effective, changed only for further clarification as work progresses. Ideas on programming are continually changing as problems are continually encountered and methods found to solve them. The final developed simulation will be completed in agreement with the ideas of this report. The aerodynamic buildup is complete. The program and data are on file at the Naval Postgraduate School, Monterey, CA. Point-of-contact is Dr. Marle D. Hewett, Department of Aeronautics (Code 67). The results of the aerodynamic buildup are verified as in agreement with tabulated and hand-calculated values. The programming though not extremely efficient, is simple, intelligible and flexible for use by various project members or even various projects.

APPENDIX A

AERODYNAMIC EQUATIONS

THE FOLLOWING ARE THE AERODYNAMIC EQUATIONS USED TO COMPUTE THE AERODYNAMIC COEFFICIENTS OF THE F/A-18 AIRCRAFT. EACH EQUATION IS GIVEN IN TERMS OF THE STATIC AND DYNAMIC COEFFICIENTS, FOLLOWED BY DEFINITIONS OF EACH TERM IN THE EQUATION. A LIST OF DERIVATIVES AND THE RESPECTIVE INDEPENDENT VARIABLES IS ALSO SHOWN FOR EACH COEFFICIENT.

THE FOLLOWING TERMS ARE USED MULTIPLE TIMES IN THE EQUATIONS

ALFA	-	ANGLE OF ATTACK
ALFACT	-	RATE OF CHANGE OF ANGLE OF ATTACK
ALTD	-	AIRCRAFT ALTITUDE
B	-	WING SPAN (AIRCRAFT REFERENCE)
BETA	-	SIDESLIP ANGLE
BETACT	-	RATE OF CHANGE OF SIDESLIP ANGLE
C	-	WING CHORD (AIRCRAFT REFERENCE)
DA	-	AVERAGE AILERON DEFLECTION
DAL/R	-	AILERON DEFLECTION (LEFT OR RIGHT)
DDA	-	DIFFERENTIAL AILERON DEFLECTION (DAL - DAR)
DDF	-	DIFFERENTIAL TRAILING EDGE FLAP DEFLECTION (DFL - DFR)
DDN	-	DIFFERENTIAL LEADING EDGE FLAP DEFLECTION (DNR - DNL)
DT	-	DIFFERENTIAL HORIZONTAL TAIL DEFLECTION (DHL - DHR)
DF	-	AVERAGE TRAILING EDGE FLAP DEFLECTION (DFL + DFR) / 2
DFL/R	-	TRAILING EDGE FLAP DEFLECTION
DH	-	AVERAGE HORIZONTAL TAIL DEFLECTION (DHL + DHR) / 2
DHL/R	-	STABILIATOR/HORIZONTAL TAIL DEFLECTION (LEFT OR RIGHT)
DN	-	AVERAGE LEADING EDGE FLAP DEFLECTION (DNL + DNR) / 2
DNL/R	-	LEADING EDGE FLAP DEFLECTION (LEFT OR RIGHT)
DR	-	AVERAGE RUDDER DEFLECTION (DRL + DRR) / 2
DRL/R	-	RUDDER DEFLECTION (LEFT OR RIGHT)
DSB	-	SPEED BRAKE DEFLECTION
LEF	-	LEADING EDGE FLAP
MACH	-	AIRCRAFT MACH NUMBER (FREE STREAM)
P	-	ROLL RATE
Q	-	PITCH RATE
QC	-	DYNAMIC PRESSURE
R	-	YAW RATE
TEF	-	TRAILING EDGE FLAP
VT	-	TOTAL AIRCRAFT VELOCITY

LONGITUDINAL AERODYNAMIC EQUATIONS

LIFT COEFFICIENT

STATIC LIFT COEFFICIENT

```
  CLST = CLBAS + ( DCLDN * DN ) + ( DCLDF * DF )
#          + ( DCLDHL + DCLDHR ) * FRCLDH / 2
#              + DCLDSB + ( DCLDAL + DCLDAR ) * FRCLDA
```

DYNAMIC LIFT COEFFICIENT

```
  CLDYN = CLQ * ( Q * C ) / ( 2 * VT )
#              + CLA * ( ALFADT * C ) / ( 2 * VT )
```

TOTAL LIFT COEFFICIENT

```
CL = CLST + CLDYN
```

WHERE:

```
CL         -  TOTAL LIFT COEFFICIENT
CLA        -  LIFT DUE TO ANGLE-OF-ATTACK RATE ( PER RAD. )
CLBAS      -  BASIC CONFIGURATION LIFT COEFFICIENT
CLDYN      -  DYNAMIC LIFT COEFFICIENT
CLQ        -  LIFT DUE TO PITCH RATE ( PER RAD. )
CLST       -  STATIC LIFT COEFFICIENT
DCLDAL/R   -  LIFT INCREMENT DUE TO AILERON DEFLECTION
              ( LEFT OR RIGHT AILERON )
DCLDF      -  LIFT INCREMENT DUE TO TEF DEFLECTION ( PER DEG. )
DCLDHL/R   -  LIFT INCREMENT DUE TO STABILATOR DEFLECTION
              ( LEFT OR RIGHT STABILATOR )
DCLDN      -  LIFT INCREMENT DUE TO LEF DEFLECTION ( PER DEG. )
DCLDSB     -  LIFT INCREMENT DUE TO SPEED BRAKE DEFLECTION
FRCLDA     -  FLEX/RIGIDITY RATIO FOR LIFT DUE TO AILERON
              DEFLECTION
FRCLDH     -  FLEX/RIGIDITY RATIO FOR LIFT DUE TO STABILATOR
              DEFLECTION
```

AND:

```
CLBAS  = F( MACH, ALTD, ALFA )
DCLCN  = F( MACH, ALTD, ALFA )
DCLCF  = F( MACH, ALTD, ALFA )
DCLDH  =  F( MACH, DH, ALFA )
FRCLCF = F( ALTD, MACH )
DCLDSB = F( MACH, DSB, ALFA )
DCLCA  = F( MACH, DA, ALFA )
FRCLCA = F( ALTD, MACH )
CLQ    = F( MACH, ALTD, ALFA )
CLA    = F( MACH, ALTD, ALFA )
```

```
DRAG COEFFICIENT

STATIC DRAG COEFFICIENT

  CDST = CDBAS + ( DCDDHL + DCDDHR ) / 2 + DCDDSB
#                + ( DCDDAL + DCDDAR ) + DCDDMF

TOTAL DRAG COEFFICIENT

CD = CDST

WHERE:

CD          -  TOTAL DRAG COEFFICIENT
CDBAS       -  BASIC CONFIGURATION DRAG COEFFICIENT
CDST        -  STATIC DRAG COEFFICIENT
DCDDAL/R    -  DRAG INCREMENT DUE TO AILERON DEFLECTION
               ( LEFT OR RIGHT AILERON )
DCDDHL/R    -  DRAG INCREMENT DUE TO STABILATOR DEFLECTION
               ( LEFT OR RIGHT STABILATOR )
DCDMF       -  DRAG INCREMENT DUE TO MANUEVERING FLAP
               DEFLECTION
DCDDSB      -  DRAG INCREMENT DUE TO SPEED BRAKE DEFLECTION

AND:

CDBAS  = F( MACH, ALFA )
DCDDH  =  F( MACH, DH, ALFA )
DCDDA  = F( MACH, DA, ALFA )
DCDDSB = F( MACH, DSB, ALFA )
DCDMF  = F( MACH, DN, ALFA )
```

```
PITCHING MOMENT COEFFICIENT


STATIC PITCHING MOMENT COEFFICIENT

CMST = CMBAS + ( DCMDN * DN ) + ( DCMDF * DF )
#              + ( DCMDHL + DCMDHR ) * FRCMDH / 2+ DCMDSB
#                  + DCMDR + ( DCMDAL + DCMDAR ) * FRCMDA


DYNAMIC PITCHING MOMENT COEFFICIENT

CMDYN = CMQ * ( Q * C ) / ( 2 * VT )
#                  + CMA * ( ALFADT * C ) / ( 2 * VT )


TOTAL PITCHING MOMENT COEFFICIENT

CM = CMST + CMDYN


WHERE:

CM         -  TOTAL PITCHING MOMENT COEFFICIENT
CMA        -  PITCHING MOMENT DUE TO ANGLE-OF-ATTACK RATE
                                                   ( PER RAD. )
CMBAS      -  BASIC CONFIGURATION PITCHING MOMENT COEFFICIENT
CMDYN      -  DYNAMIC PITCHING MOMENT COEFFICIENT
CMQ        -  PITCHING MOMENT DUE TO PITCH RATE ( PER RAD. )
CMST       -  STATIC PITCHING MOMENT COEFFICIENT
DCMDAL/R   -  PITCHING MOMENT INCREMENT DUE TO AILERON
              DEFLECTION ( LEFT OR RIGHT AILERON )
DCMDF      -  PITCHING MOMENT INCREMENT DUE TO TEF DEFLECTION
                                                   ( PER DEG. )
DCMDHL/R   -  PITCHING MOMENT INCREMENT DUE TO STABILATOR
              DEFLECTION ( LEFT OR RIGHT STABILATOR )
DCMDN      -  PITCHING MOMENT INCREMENT DUE TO LEF DEFLECTION
                                                   ( PER DEG. )
DCMDR      -  PITCHING MOMENT INCREMENT DUE TO RUDDER DEFLECTION
DCMDSB     -  PITCHING MOMENT INCREMENT DUE TO SPEED BRAKE
              DEFLECTION
FRCMDA     -  FLEX/RIGIDITY RATIO FOR PITCHING MOMENT DUE TO
              AILERON DEFLECTION
FRCMDH     -  FLEX/RIGIDITY RATIO FOR PITCHING MOMENT DUE TO
              STABILATOR DEFLECTION
```

```
AND:

CMBAS  = F( MACH, ALTD, ALFA )
DCMCN  = F( MACH, ALTD, ALFA )
DCMCF  = F( MACH, ALTD, ALFA )
DCMCH  = F( MACH, DH, ALFA )
FRCMCH = F( ALTD, MACH )
DCMCSB = F( MACH, DSB, ALFA )
DCMDR  = F( MACH, DR, ALFA )
DCMCA  = F( MACH, DA, ALFA )
FRCMCA = F( ALTD, MACH )
CMQ    = F( MACH, ALTD, ALFA )
CMA    = F( MACH, ALTD, ALFA )
```

LATERAL-DIRECTIONAL AERODYNAMIC EQUATIONS

YAWING MOMENT COEFFICIENT

STATIC YAWING MOMENT COEFFICIENT

```
  CNST = CNBAS + ( DCNBFX * BETA ) + DCNDN + DCNDF
#            + ( DCNDAL + DCNDAR ) * FRCNDA
#            + ( KRDR * DCNDR * FRCNDR )
#            + ( DCNDT * FRCNDT * DT ) + ( DCNDSB * BETA )
```

DYNAMIC YAWING MOMENT COEFFICIENT

```
  CNDYN = ( CNR + DCNRFX )  * ( R * B ) / ( 2 * VT )
#             + ( CNP * FRCNP ) * ( P * B ) / ( 2 * VT )
```

TOTAL YAWING MOMENT COEFFICIENT

```
CN = CNST + CNDYN
```

WHERE:

```
CN        -  TOTAL YAWING MOMENT COEFFICIENT
CNBAS     -  BASIC CONFIGURATION YAWING MOMENT COEFFICIENT
CNDYN     -  DYNAMIC YAWING MOMENT COEFFICIENT
CNP       -  YAWING MOMENT DUE TO ROLL RATE
CNR       -  YAWING MOMENT DUE TO YAW RATE
CNST      -  STATIC YAWING MOMENT COEFFICIENT
DCNBFX    -  YAWING MOMENT FLEXIBILITY DERIVATIVE DUE TO
             SIDESLIP
DCNDAL/R  -  YAWING MOMENT INCREMENT DUE TO AILERON DEFLECTION
             ( LEFT OR RIGHT AILERON )
DCNDF     -  YAWING MOMENT INCREMENT DUE TO TEF DEFLECTION
                                                  ( PER DEG. )
DCNDN     -  YAWING MOMENT INCREMENT DUE TO LEF DEFLECTION
                                                  ( PER DEG. )
DCNDR     -  YAWING MOMENT INCREMENT DUE TO RUDDER DEFLECTION
DCNDSB    -  YAWING MOMENT INCREMENT DUE TO SPEED BRAKE
             DEFLECTION
DCNDT     -  YAWING MOMENT INCREMENT DUE TO DIFFERENTIAL TAIL
             DEFLECTION
DCNRFX    -  YAWING MOMENT INCREMENT DUE TO FLEXIBILITY DUE TO
```

```
         SIDESLIP
FRCNCA   - FLEX/RIGIDITY RATIO FOR YAWING MOMENT DUE TO
           AILERON DEFLECTION
FRCNCR   - FLEX/RIGIDITY RATIO FOR YAWING MOMENT DUE TO
           RUDDER DEFLECTION
FRCNCT   - FLEX/RIGIDITY RATIO FOR YAWING MOMENT DUE TO
           DIFFERENTIAL HORIZONTAL TAIL
FRCNP    - FLEX/RIGIDITY RATIO FOR YAWING MOMENT DUE TO
           ROLL RATE
KRDR     - RUDDER POWER FACTOR DUE TO STABILATOR POSITION

AND:

CNBAS  = F( MACH, ALFA, BETA )
DCNCN  = F( MACH, DN, ALFA, BETA )
DCNDF  = F( MACH, DF, ALFA, BETA )
DCNCT  = F( MACH, DH, ALFA )
FRCNCT = F( MACH, ALTD, ALFA )
DCNCSB = F( MACH, DSB, BETA, ALFA )
DCNCR  = F( MACH, DR, BETA, ALFA )
FRCNDR = F( ALTD, MACH )
DCNCA  = F( MACH, DA, ALFA )
FRCNCA = F( ALTD, MACH )
CNR    = F( MACH, ALFA )
CNP    = F( MACH, ALFA )
DCNRFX = F( ALTD, MACH )
FRCNP  = F( MACH, ALFA )
DCNBFX = F( MACH, QC, ALFA )
KRDR   = F( CH, ALFA )
```

```
ROLLING MOMENT COEFFICIENT

STATIC ROLLING MOMENT COEFFICIENT

CRST = CRBAS + ( DCRBFX * BETA ) + DCRDN + DCRDF
#          + ( DCRDAL + DCRDAR ) * FRCRDA + ( DCRDR * FRCRDR )
#          + ( DCRDT * FRCRDT * DT ) + ( DCRDSB * BETA )
#          + DCRASY + ( CRDDF * DDF ) + ( CRDDN * DDN )

DYNAMIC ROLLING MOMENT COEFFICIENT

CRDYN = ( CRR + DCRRFX ) * ( R * B ) / ( 2 * VT )
#             + ( CRP ) * ( P * B ) / ( 2 * VT )

TOTAL ROLLING MOMENT COEFFICIENT

CR = CRST + CRDYN


WHERE:

CR         -  TOTAL ROLLING MOMENT COEFFICIENT
CRBAS      -  BASIC CONFIGURATION ROLLING MOMENT COEFFICIENT
CRDDN      -  ROLLING MOMENT INCREMENT DUE TO DIFFERENTIAL LEF
              DEFLECTION < PER DEG. >
CRDDF      -  ROLLING MOMENT INCREMENT DUE TO DIFFERENTIAL TEF
              DEFLECTION OR FLAPERON DEFLECTION < PER DEG. >
CRDYN      -  DYNAMIC ROLLING MOMENT COEFFICIENT
CRP        -  ROLLING MOMENT DUE TO ROLL RATE
CRR        -  ROLLING MOMENT DUE TO YAW RATE
CRST       -  STATIC ROLLING MOMENT COEFFICIENT
DCRASY     -  ROLLING MOMENT INCREMENT DUE TO NOSE ASYMETRY
DCRBFX     -  ROLLING MOMENT FLEXIBILITY DERIVATIVE DUE TO
              SIDESLIP
DCRDAL/R   -  ROLLING MOMENT INCREMENT DUE TO AILERON DEFLECTION
              ( LEFT OR RIGHT AILERON )
DCRDF      -  ROLLING MOMENT INCREMENT DUE TO LEF DEFLECTION
DCRDN      -  ROLLING MOMENT INCREMENT DUE TO TEF DEFLECTION
DCRDR      -  ROLLING MOMENT INCREMENT DUE TO RUDDER DEFLECTION
DCRDSB     -  ROLLING MOMENT INCREMENT DUE TO SPEEDBRAKE
              DEFLECTION
DCRDT      -  ROLLING MOMENT INCREMENT DUE TO DIFFERENTIAL TAIL
              DEFLECTION
DCRRFX     -  ROLLING MOMENT INCREMENT DUE TO FLEXIBILITY DUE TO
              SIDESLIP
FRCRDA     -  FLEX/RIGIDITY RATIO FOR ROLLING MOMENT DUE TO
              AILERON DEFLECTION
```

```
FRCRDR    -   FLEX/RIGIDITY RATIO FOR ROLLING MOMENT DUE TO
              RUDDER DEFLECTION
FRCRDT    -   FLEX/RIGIDITY RATIO FOR ROLLING MOMENT DUE TO
              DIFFERENTIAL HORIZONTAL TAIL DEFLECTION


AND:

CRBAS  = F( MACH, ALFA, BETA )
DCRDN  = F( MACH, DN, ALFA, BETA )
DCRDF  = F( MACH, DF, ALFA, BETA )
DCRDT  = F( MACH, DH, ALFA )
FRCRDT = F( ALTD, MACH )
DCRDSB = F( MACH, DSB, BETA, ALFA )
DCRDR  = F( MACH, DR, BETA, ALFA )
FRCRDR = F( ALTD, MACH )
DCRDA  = F( MACH, DA, ALFA )
FRCRDA = F( ALTD, MACH )
CRR    = F( MACH, ALFA )
CRP    = F( MACH, ALTD, ALFA )
DCRRFX = F( ALTD, MACH )
DCRASY = F( ALFA )
DCRBFX = F( MACH, QC, ALFA )
CRDDN  = F( MACH, QC )
CRDDF  = F( MACH, ALTD, ALFA )
```

```
SIDE FORCE COEFFICIENT

STATIC SIDE FORCE COEFFICIENT

CYST = CYBAS + ( DCYBFX * BETA ) + DCYDN + DCYDF
#         + ( DCYDAL + DCYDAR ) * FRCYDA + ( DCYDR * FRCYDR )
#         + ( DCYDT * FRCYDT * DT ) + ( DCYDSB * BETA )

DYNAMIC SIDE FORCE COEFFICIENT

CYDYN = ( CYR + DCYRFX )  * ( R * B ) / ( 2 * VT )
#          + ( CYP * FRCYP ) * ( P * B ) / ( 2 * VT )

TOTAL SIDE FORCE COEFFICIENT

CY = CYST + CYDYN


WHERE:

CY         -  TOTAL SIDE FORCE COEFFICIENT
CYBAS      -  BASIC CONFIGURATION SIDE FORCE COEFFICIENT
CYDYN      -  DYNAMIC SIDE FORCE COEFFICIENT
CYP        -  SIDE FORCE DUE TO ROLL RATE
CYR        -  SIDE FORCE DUE TO YAW RATE
CYST       -  STATIC SIDE FORCE COEFFICIENT
DCYBFX     -  SIDE FORCE FLEXIBILITY DERIVATIVE DUE TO
              SIDESLIP
DCYDAL/R   -  SIDE FORCE INCREMENT DUE TO AILERON DEFLECTION
              ( LEFT OR RIGHT AILERON )
DCYDN      -  SIDE FORCE INCREMENT DUE TO LEF DEFLECTION
DCYDF      -  SIDE FORCE INCREMENT DUE TO TEF DEFLECTION
DCYDR      -  SIDE FORCE INCREMENT DUE TO RUDDER DEFLECTION
DCYDSB     -  SIDE FORCE INCREMENT DUE TO SPEEDBRAKE DEFLECTION
DCYDT      -  SIDE FORCE INCREMENT DUE TO DIFFERENTIAL TAIL
              DEFLECTION
DCYRFX     -  SIDE FORCE INCREMENT DUE TO FLEXIBILITY DUE TO
              SIDESLIP
FRCYDA     -  FLEX/RIGIDITY RATIO FOR SIDE FORCE DUE TO AILERON
              DEFLECTION
FRCYDR     -  FLEX/RIGIDITY RATIO FOR SIDE FORCE DUE TO RUDDER
              DEFLECTION
FRCYDT     -  FLEX/RIGIDITY RATIO FOR SIDE FORCE DUE TO
              DIFFERENTIAL HORIZONTAL DEFLECTION
FRCYP      -  FLEX/RIGIDITY RATIO FOR SIDE FORCE DUE TO ROLL
              RATE
```

AND:

```
CYBAS  = F( MACH, ALFA, BETA )
DCYDN  = F( MACH, DN, ALFA, BETA )
DCYDF  = F( MACH, DF, ALFA, BETA )
DCYDT  = F( MACH, DH, ALFA )
FRCYDT = F( ALTD, MACH )
DCYDSB = F( MACH, DSB, BETA, ALFA )
DCYDR  = F( MACH, DR, BETA, ALFA )
FRCYDR = F( ALTD, MACH )
DCYDA  = F( MACH, DA, ALFA )
FRCYDA = F( ALTD, MACH )
CYR    = F( MACH, ALFA )
CYP    = F( MACH, ALFA )
DCYRFX = F( ALTD, MACH )
FRCYF  = F( ALTD, MACH )
CYBFLX = F( ALTD, MACH )
```

APPENDIX B

REFERENCE FLIGHT CONDITIONS

INDEPENDENT VARIABLE TABULATED VALUES

REFERENCE ANGLE OF ATTACK VALUES - LONGITUDINAL DATA

-4.0	0.0	4.0	8.0	12.0	16.0	20.0	24.0
28.0	32.0	36.0	40.0	45.0	50.0	55.0	60.0
65.0	70.0	75.0	80.0	85.0	90.0		

REFERENCE ANGLE OF ATTACK VALUES - LATERAL-DIRECTIONAL DATA

0.0	10.0	15.0	20.0	25.0	30.0	35.0	40.0
45.0	50.0	55.0	60.0	65.0	70.0	75.0	80.0
85.0	90.0						

REFERENCE ALTITUDE VALUES

0.0	20000.0	40000.0	60000.0

REFERENCE SIDESLIP ANGLE VALUES

-20.0	-16.0	-12.0	-8.0	-4.0	0.0	4.0	8.0
12.0	16.0	20.0					

REFERENCE AILERON DEFLECTION VALUES

-25.0	-12.5	0.0	12.5	25.0

REFERENCE T.E. FLAP DEFLECTION VALUES

0.0 20.0

REFERENCE HORIZ. TAIL DEFLECTION VALUES

-24.0 -12.0 -6.0 0.0 6.0 10.5

REFERENCE L.E. FLAP DEFLECTION VALUES

0.0 25.0

REFERENCE MANEUVERING FLAP < LEF > VALUES

0.0 6.0 15.0 34.0

REFERENCE RUDDER DEFLECTION VALUES

-30.0 0.0 30.0

REFERENCE SPEED BRAKE DEFLECTION VALUES

0.0 60.0

REFERENCE DYNAMIC PRESSURE VALUES

0.0	2000.0

REFERENCE MACH NUMBER VALUES

0.2	0.6	0.8	0.9

ATMOSPHERIC TABLE ALTITUDE VALUES

0.0	1000.0	2000.0	3000.0	4000.0	5000.0	6000.0	7000.0
8000.0	9000.0	10000.0	11000.0	12000.0	13000.0	14000.0	15000.0
16000.0	17000.0	18000.0	19000.0	20000.0	21000.0	22000.0	23000.0
24000.0	25000.0	26000.0	27000.0	28000.0	29000.0	30000.0	31000.0
32000.0	33000.0	34000.0	35000.0	36000.0	40000.0	45000.0	50000.0
55000.0	60000.0	65000.0					

APPENDIX C

SAMPLE PROGRAM STATEMENTS

```
EXAMFLE: THIS IS A LISTING OF THE AERODYNAMIC BUILDUP FOR
         LIFT COEFFICIENT ONLY.  THE FULL AERODYNAMIC BUILDUP
         FOLLOWS THE SAME FORMAT EXPANDED IN SECTIONS ONE, TWO,
         FOUR AND FIVE FOR THE REMAINING COEFFICIENTS.
         SECTION THREE IS THE SAME AS THE FULL
         BUILDUP. THE SUBROUTINES HAVE BEEN LISTED IN
         ANOTHER APPENDIX.

THIS FROGRAM PERFORMS THE AERODYNAMIC BUILD-UP FOR THE
F/A-18A FIGHTER ATTACK AIRCRAFT.  TABULATED AERODYNAMIC DATA
EXTRACTEL FROM GRAPHICAL PRESENTAIONS IS REFERENCED USING
INTERPCLATION ROUTINES FOR INTERMMECIATE AND TABULATED FLIGHT
CONCITIONS.  THE AIRCRAFT TOTAL COEFFICIENTS FOR LIFT, DRAG,
PITCHING MOMENT, ROLLING MOMENT, YAWING MOMENT, AND SIDE
FORCE ARE DETERMINED USING THE STATIC AND DYNAMIC DERIVATIVE
DATA, ALCNG WITH ADDITIONAL REQUIRED AERCDYNAMIC CONTROL
FACTCRS.  THE AERODYNAMIC CCEFFICIENTS CAN BE DETERMINED
THROUGH THE FOLLOWING RANGE OF FLIGHT CONDITIONS:

      MACH NUMBER:                .2  -  .9
      ALTITUDE:                   SEA LEVEL  -  60,000 FT
      ANGLE-CF-ATTACK:            (-)4  -  90 DEG
      SIDESLIP ANGLE:             (-)20  -  2C DEG
      DYNAMIC PRESSURE:           4.2  -  1200.8

THE FROGRAM PROVIDES RESULTS OF THE COEFFICIENT BUILDUP FOR
INDEPENDENT STUDY OR FOR INTEGRATION INTC FLIGHT SIMULATION
PROGRAMS.

THE FROGRAM IS SEGMENTED INTO SIX MAJOR SECTIONS AS FOLLOWS:

     SECTION ONE:      VARIABLE DEFINITION, EXPLANATIONS,
                       DECLARATICNS, AND FROGRAM PARAMETERS
     SECTION TWO:      AERODYNAMIC DATA AND CONSTANTS
     SECTION THREE:    TEST FLIGHT CONDITCN INPUTS
     SECTION FOUR:     AERODYNAMIC BUILD-UP
     SECTION FIVE:     OUTPUT ANC CONTROL
     SECTION SIX:      SUBROUTINES

EACH SECTION IS FURTHER DIVIDED INTC MULTIPLE SUBSECTIONS
AS RECUIRED AND INDICATED IN THE PRCGRAM COMMENTS.

PROGRAMMER IS:  LT. A.L. RAITHEL, H-232, EXT. 2866
```

```
SECTICN 1:  DEFINITIONS, EXPLANATIONS, DECLARATICNS AND
            PROGRAM PARAMETERS

( )  ALFA       -  ANGLE OF ATTACK
( )  ALFACT     -  RATE OF CHANGE OF ANGLE OF ATTACK
( )  ALTC       -  AIRCRAFT ALTITUDE
( )  ATMOS1     -  STANDARD DAY DENSITY TABLE
( )  ATMOS2     -  STANDARD DAY SONIC VELOCITY TABLE
( )  B          -  WING SPAN ( AIRCRAFT REFERENCE )
( )  BETA       -  SIDESLIP ANGLE
( )  BETACT     -  RATE OF CHANGE OF SIDESLIP ANGLE
( )  C          -  WING CHORD ( AIRCRAFT REFERENCE )
( )  CL         -  TOTAL LIFT COEFFICIENT
( )  CLA        -  LIFT DUE TO ANGLE-OF-ATTACK RATE ( PER RAD. )
( )  CLBAS      -  BASIC CONFIGURATION LIFT COEFFICIENT
( )  CLDYN      -  DYNAMIC LIFT COEFFICIENT
( )  CLOUT      -  CONTROL VARIABLE FOR OUTPUT OF LIFT COEFFICIENT
                   DERIVATIVES
( )  CLQ        -  LIFT DUE TO PITCH RATE ( PER RAD. )
( )  CLST       -  STATIC LIFT COEFFICIENT
( )  CL1        -  CLBAS DATA TABLE
( )  CL2        -  DCLDN DATA TABLE
( )  CL3        -  DCLDF DATA TABLE
( )  CL4        -  DCLDH DATA TABLE
( )  CL5        -  FRCLDH DATA TABLE
( )  CL6        -  DCLDSB DATA TABLE
( )  CL7        -  DCLDA DATA TABLE
( )  CL8        -  FRCLDA DATA TABLE
( )  CL9        -  CLQ DATA TABLE
( )  CL10       -  CLA DATA TABLE
( )  DAL/R      -  AILERON DEFLECTION ( LEFT OR RIGHT )
( )  DCLDAL/R   -  LIFT INCREMENT DUE TO AILERON DEFLECTION
                   ( LEFT OR RIGHT AILERON )
( )  DCLDF      -  LIFT INCREMENT DUE TO TEF DEFLECTION ( PER DEG. )
( )  DCLDHL/R   -  LIFT INCREMENT DUE TO STABILATOR DEFLECTION
                   ( LEFT OR RIGHT STABILATOR )
( )  DCLDN      -  LIFT INCREMENT DUE TO LEF DEFLECTION ( PER DEG. )
( )  DCLDSB     -  LIFT INCREMENT DUE TO SPEED BRAKE DEFLECTION
( )  DDA        -  DIFFERENTIAL AILERON DEFLECTION ( DAL - DAR )
( )  DDF        -  DIFFERENTIAL TRAILING EDGE FLAP DEFLECTION
                                                    ( DFL - DFR )
( )  DDN        -  DIFFERENTIAL LEADING EDGE FLAP DEFLECTION
                                                  ( DNR - DNL )
( )  DT         -  DIFFERENTIAL HORIZONTAL TAIL DEFLECTION
                                            ( DHL - DHR )
( )  DF         -  AVERAGE TRAILING EDGE FLAP DEFLECTION
```

```
                                            ( DFL + DFR ) / 2
( )  DFL/R      -  TRAILING EDGE FLAP DEFLECTION
( )  DH         -  AVERAGE HORIZONTAL TAIL DEFLECTION
                                            ( DHL + DHR ) / 2
( )  DHL/R      -  STABILIATOR/HORIZONTAL TAIL DEFLECTION ( LEFT
                   OR RIGHT )
( )  DN         -  AVERAGE LEADING EDGE FLAP DEFLECTION
                                            ( DNL + DNR ) / 2
( )  DNL/R      -  LEADING EDGE FLAP DEFLECTION ( LEFT OR RIGHT )
( )  DR         -  AVERAGE RUDDER DEFLECTION ( DRL + DRR ) / 2
( )  DRL/R      -  RUDDER DEFLECTION ( LEFT OR RIGHT)
( )  DSB        -  SPEED BRAKE DEFLECTION
( )  FORVAR     -  FOUR VARIABLE INTERPOLATION SUBROUTINE
( )  FRCLCA     -  FLEX/RIGIDITY RATIO FOR LIFT DUE TO AILERON
                   DEFLECTION
( )  FRCLCH     -  FLEX/RIGIDITY RATIO FOR LIFT DUE TO STABILATOR
                   DEFLECTION
( )  HCAD       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   ATMOSPHERIC DATA
( )  HCCD       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   DRAG COEFFICIENT DERIVATIVE DATA
( )  HCCL       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   LIFT COEFFICIENT DERIVATIVE DATA
( )  HCCM       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   PITCHING MOMENT COEFFICIENT DERIVATIVE DATA
( )  HCCN       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   YAWING MOMENT COEFFICIENT DERIVATIVE DATA
( )  HCCR       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   ROLLING MOMENT COEFFICIENT DERIVATIVE DATA
( )  HCCY       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   SIDE FORCE COEFFICIENT DERIVATIVE DATA
( )  HCFC       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   FLIGHT CONDITION INPUTS
( )  HCIV       -  CONTROL VARIABLE FOR HARDCOPY OUTPUT OF TABULATED
                   FLIGHT CONDITION INDEPENDENT VARIABLES
( )  I          -  INTEGER VALUE FOR SUBSCRIPTING
( )  IVALF1()   -  INDEPENDENT VARIABLE ALFA VALUES FOR WHICH
                   LONGITUDINAL DATA IS TABULATED
( )  IVALF2()   -  INDEPENDENT VARIABLE ALFA VALUES FOR WHICH
                   LATERAL-DIRECTIONAL DATA IS TABULATED
( )  IVALTD()   -  INDEPENDENT VARIABLE ALTITUDE VALUES FOR WHICH
                   DATA IS TABULATED
( )  IVBETA()   -  INDEPENDENT VARIABLE BETA VALUES FOR WHICH DATA
                   IS TABULATED
( )  IVDA()     -  INDEPENDENT VARIABLE AILERON DEFLECTION VALUES
                   FOR WHICH DATA IS TABULATED
( )  IVDF()     -  INDEPENDENT VARIABLE TEF DEFLECTION VALUES FOR
                   WHICH DATA IS TABULATED
```

```
( )  IVDH()    -  INDEPENDENT VARIABLE HORIZ. TAIL DEFLECTION
                  VALUES FOR WHICH DATA IS TABULATED
( )  IVDN()    -  INDEPENDENT VARIABLE LEF DEFLECTION VALUES FOR
                  WHICH DATA IS TABULATED
( )  IVDR()    -  INDEPENDENT VARIABLE RUDDER DEFLECTION VALUES
                  FOR WHICH DATA IS TABULATED
( )  IVDSE()   -  INDEPENDENT VARIABLE SPEED BRAKE DEFLECTION
                  VALUES FOR WHICH DATA IS TABULATED
( )  IVQC()    -  INDEPENDENT VARIABLE DYNAMIC PRESSURE VALUES
                  FOR WHICH DATA IS TABULATED
( )  IVMACH()  -  INDEPENDENT VARIABLE MACH VALUES FOR WHICH DATA
                  IS TABULATED
( )  IVMF()    -  INDEPENDENT VARIABLE MANEUVERING FLAP LEF
                  DEFLECTIONS VALUES FOR WHICH DATA IS TABULATED
( )  J         -  INTEGER VALUE FOR SUBSCRIPTING
( )  K         -  INTEGER VALUE FOR SUBSCRIPTING
( )  KRDR      -  RUDDER POWER FACTOR DUE TO STABILATOR POSITION
( )  L         -  INTEGER VALUE FOR SUBSCRIPTING
( )  L.E.D.    -  LEADING EDGE DOWN
( )  LEF       -  LEADING EDGE FLAP
( )  L.E.L.    -  LEADING EDGE LEFT
( )  MACH      -  AIRCRAFT MACH NUMBER ( FREE STREAM )
( )  ONEVAR    -  ONE VARIABLE INTERPOLATION SUBROUTINE
( )  P         -  ROLL RATE
( )  Q         -  PITCH RATE
( )  QC        -  DYNAMIC PRESSURE
( )  R         -  YAW RATE
( )  RHO       -  ATMOSPHERIC DENSITY
( )  STDALT()  -  ALTITUDE TABLE FOR STANDARD DAY ATMOSPHERIC DATA
( )  SVEL      -  SONIC VELOCITY
( )  TAC       -  CONTROL VARIABLE FOR INPUT OF TEST ATMOSPHERIC
                  DATA
( )  TCSD      -  CONTROL VARIABLE FOR INPUT OF TEST CONTROL
                  SURFACE DEFLECTIONS
( )  TFC       -  CONTROL VARIABLE FOR INPUT OF TEST FLIGHT
                  CONDITION PARAMETERS
( )  T.E.D.    -  TRAILING EDGE DOWN
( )  TEF       -  TRAILING EDGE FLAP
( )  THRVAR    -  THREE VARIABLE INTERPOLATION SUBROUTINE
( )  TUVAR     -  TWO VARIABLE INTERPOLATION SUBROUTINE
( )  VT        -  TOTAL AIRCRAFT VELOCITY
( )  WSMX      -  WORKING SPACE MATRIX FOR SUBROUTINES ( ONE
                  DIMENSIONAL )
( )  WSMXY     -  WORKING SPACE MATRIX FOR SUBROUTINES ( TWO
                  DIMENSIONAL )
( )  WSMXYZ    -  WORKING SPACE MATRIX FOR SUBROUTINES ( THREE
                  DIMENSIONAL )
```

```
PROGRAM OPERATION NOTES:

(1)   THE FOLLOWING FILE DEFINITIONS APPLY TO THIS PROGRAM
      FOR TRANSFER OF DATA BETWEEN THE PROGRAM DATA TABLES
      AND THE DATA FILES

        FILEDEF 01 DISK F18IV DATA A  -  IV DATA
        FILEDEF 02 DISK F18CL DATA A  -  CL DATA
        FILEDEF 03 DISK F18CD DATA A  -  CD DATA
        FILEDEF 04 DISK F18CM DATA A  -  CM DATA
        FILEDEF 07 DISK F18CN DATA A  -  CN DATA
        FILEDEF 08 DISK F18CR DATA A  -  CR DATA
        FILEDEF 09 DISK F18CY DATA A  -  CY DATA
        FILEDEF 10 DISK ATMOS DATA A  -  ATMOSPHERIC DATA

(2)   THE PROGRAM AERODYNAMIC DATA TABLES ARE AUTOMATICALLY
      LOADED FROM THE DATA FILES FOR THE AERODYNAMIC BUILDUP
      CALCULATIONS.  IF A HARDCOPY OF A DATA TABLE IS DESIRED
      SET THE APPROPRIATE PARAMETER EQUAL TO ONE.  IF NO
      HARD COPY IS DESIRED SET THE PARAMETER EQUAL TO ZERO.
      I.E.  IF A HARDCOPY OF THE TABULATED CL DATA IS
      DESIRED SET HCCL/1/, IF NO HARDCOPY OF THE CY DATA IS
      DESIRED SET HCCY/0/.  SET THE PARAMETERS IN SECTION ONE
      PROGRAM CONTROL DATA CARDS.

(3)   IF A HARDCOPY OF THE OUTPUT AERODYNAMIC COEFFICIENT
      DERIVATIVES IS DESIRED, SET THE APPROPRIATE PARAMETER
      EQUAL TO ONE.  IF NO HARDCOPY IS DESIRED, SET THE
      PARAMETER EQUAL TO ZERO.  I.E., IF A HARDCOPY OF THE
      OUTPUT CL DERIVATIVES IS DESIRED SET CLOUT/1/, IF
      NO HARDCOPY OF CN DERIVATIVES IS DESIRED, SET CNOUT/0/.
      SET THE PARAMETERS IN SECTION ONE PROGRAM CONTROL DATA
      CARDS.

(4)   IF A PROGRAM RUN REQUIRES TEST FLIGHT CONDITION INPUTS
      SET THE APPROPRIATE PARAMETER EQUAL TO ONE.  IF THE
      INPUTS ARE BEING PROVIDED BY A FLIGHT SIMULATION
      SET THE PARAMETER EQUAL TO ZERO.  I.E., IF THE VALUES
      FOR FLIGHT CONDITIONS, ALFA, MACH, Q, ETC., ARE
      GENERATED IN A MAIN PROGRAM, SET TFC/0/.  IF THE
      VALUES OF DYNAMIC PRESSURE, VELOCITY, ETC., ARE NOT
      PROVIDED, SET TAC/1/, TO UTILIZE THE INCORPORATED
      ATMOSPHERIC DATA TABLES.

(5)   THIS AERODYNAMIC BUILDUP USES THE FOLLOWING SIGN
      CONVENTIONS FOR CONTROL SURFACE DEFLECTIONS

             DHL/R  -  POSITIVE T.E.D.
```

```
        DNL/R  -  POSITIVE L.E.D.
        DFL/R  -  POSITIVE T.E.D.
        DAL/R  -  POSITIVE T.E.D.
        DRL/R  -  POSITIVE T.E.L.

(6)  DATA IN THIS PROGRAM IS READ AND WRITTEN BY ROWS, WITH
     THE RIGHT MOST ARGUMENT INCREMENTING MOST OFTEN. I.E.
     CLEAS = F(MACH, ALTD, ALFA) = CL1(4, 4, 22) IS READ AND
     WRITTEN,  (1,1,1), (1,1,2), ..., (1,1,8),
               (1,1,9), ..., (1,1,22), (1,2,1), (1,2,2),
               ..., (1,2,22), (1,3,1), ..., (1,3,22), (1,4,1),
               ..., (1,4,22), (2,1,1), ..., (2,1,22), ...,
               (2,2,1), ..., ETC.
     ALL NUMBERS IN THE PROGRAM NOT DECLARED ARE REAL NUMBERS
```

DIMENSION/DECLARATION STATEMENT

```
REAL B, C, ALFA, BETA, ALTD, MACH, QC, Q, ALFADT, VT, P, R, DSB,
#     DAL, DAR, DDA, DFL, DFR, DF, DNL, DNR, DN, DRL, DRR, DR,
#     DHL, DHR, DH, DT, KRDR, STDALT(43), ATMOS1(43), ATMOS2(43),
#     IVALF1(22), IVALF2(18), IVALTD(4), IVBETA(11), IVDA(5),
#     IVDF(2), IVDH(6), IVDN(2), IVDR(3), IVDSB(2), IVQC(2),
#     IVMF(4), IVMACH(4), WSMXYZ(5,16,22), WSMXY(5,16), WSMX(22)

DIMENSION  CD1(4,22), CD2(4,6,22), CD3(4,5,22), CD4(4,2,22),
#     CD5(4,4,22),
#     CL1(4,4,22), CL2(4,4,22), CL3(4,4,22), CL4(4,6,22),
#     CL5(4,4), CL6(4,2,22), CL7(4,5,22), CL8(4,4), CL9(4,4,22),
#     CL10(4,4,22), CM1(4,4,22), CM2(4,4,22), CM3(4,4,22),
#     CM4(4,6,22), CM5(4,4), CM6(4,2,22), CM7(4,3,22), CM8(4,5,22),
#     CM9(4,4), CM10(4,4,22), CM11(4,4,22),
#     CN1(4,18,11), CN2(4,2,18,11), CN3(4,2,18,11), CN4(4,6,18),
#     CN5(4,4,18), CN6(4,2,11,18), CN7(4,3,11,18), CN8(4,4),
#     CN9(4,5,18), CN10(4,4), CN11(4,18), CN12(4,18), CN13(4,4),
#     CN14(4,18), CN15(4,2,18), CN16(6,18),
#     CR1(4,18,11), CR2(4,2,18,11), CR3(4,2,18,11), CR4(4,6,18),
#     CR5(4,4), CR6(4,2,11,18), CR7(4,3,11,18), CR8(4,4),
#     CR9(4,5,18), CR10(4,4), CR11(4,18), CR12(4,4,18), CR13(4,4),
#     CR14(18), CR15(4,2,18), CR16(4,2), CR17(4,4,18),
```

```
#     CY1(4,18,11), CY2(4,2,18,11), CY3(4,2,18,11), CY4(4,6,18),
#     CY5(4,4), CY6(4,2,11,18), CY7(4,3,11,18), CY8(4,4),
#     CY9(4,5,18), CY10(4,4), CY11(4,18), CY12(4,18), CY13(4,4),
#     CY14(4,4), CY15(4,4)

 INTEGER I, J, K, L,
#         HCAC, HCCD, HCCL, HCCM, HCCN, HCCR, HCCY, HCFC, HCIV,
#         CDOUT, CLOUT, CMOUT, CNOUT, CROUT, CYOUT,
#         TAC, TCSD, TFC


    PROGRAM CONTROL DATA

 DATA   HCAD/0/, HCCD/0/, HCCL/1/, HCCM/0/, HCCN/0/, HCCR/0/,
#       HCCY/0/, HCFC/1/, HCIV/0/
 DATA   CDOUT/0/, CLOUT/1/, CMOUT/0/, CNOUT/0/,
#       CROUT/0/, CYOUT/0/
 DATA   TAC/1/, TCSD/1/, TFC/1/
```

```
SECTION 2:  AERODYNAMIC DATA AND CONSTANTS

THIS SECTION READS/LOADS THE AERODYNAMIC DATA FROM THE DATA
FILES INTO THE APPROPRIATE PROGRAM DATA TABLE AND IF DESIRED
PROVIDES A HARDCOPY OF DATA TABLES.  REFER TO THE NOTES IN
SECTION ONE FOR VERIFICATION OF HARDCOPY OUTPUT.

AIRCRAFT ASSOCIATED CONSTANTS

DATA   B/37.42/,     C/11.52/


AERODYNAMIC DATA

INDEPENDENT VARIABLE DATA

ANGLE OF ATTACK - IVALF1

ANGLE OF ATTACK - IVALF2

ALTITUDE - IVALTD

SIDESLIP ANGLE - IVBETA

AILERON DEFLECTION - IVDA

TRAILING EDGE FLAP DEFLECTON - IVDF

HORIZONTAL TAIL DEFLECTION - IVDH

LEADING EDGE FLAP DEFLECTION - IVDN

RUDDER DEFLECTION - IVDR

SPEED BRAKE DEFLECTION - IVDSB

DYNAMIC PRESSURE - IVQC

MACH NUMBER - IVMACH

STANDARD ALTITUDE - STDALT

MANEUVERING FLAP DEFLECTION ( LEF ) - IVMF
```

```
READ(1,100) ( IVALF1(I), I = 1,22 )
READ(1,100) ( IVALF2(I), I = 1,18 )
READ(1,100) ( IVALTD(I), I = 1,4 )
READ(1,100) ( IVBETA(I), I = 1,11 )
READ(1,100) ( IVDA(I), I = 1,5 )
READ(1,100) ( IVDF(I), I = 1,2 )
READ(1,100) ( IVDH(I), I = 1,6 )
READ(1,100) ( IVDN(I), I = 1,2 )
READ(1,100) ( IVDR(I), I = 1,3 )
READ(1,100) ( IVDSB(I), I = 1,2 )
READ(1,100) ( IVQC(I), I = 1,2 )
READ(1,100) ( IVMACH(I), I = 1,4 )
READ(1,100) ( STDALT(I), I = 1,43 )
READ(1,100) ( IVMF(I), I = 1,4 )

IF ( HCIW .EQ. 0 ) GO TO 101
WRITE(6,105)
WRITE(6,110)
WRITE(6,120) ( IVALF1(I), I = 1,22 )
WRITE(6,112)
WRITE(6,120) ( IVALF2(I), I = 1,18 )
WRITE(6,130)
WRITE(6,120) ( IVALTD(I), I = 1,4 )
WRITE(6,140)
WRITE(6,120) ( IVBETA(I), I = 1,11 )
WRITE(6,150)
```

```
      WRITE(6,120) ( IVDA(I), I = 1,5 )

      WRITE(6,160)
      WRITE(6,120) ( IVDF(I), I = 1,2 )

      WRITE(6,170)
      WRITE(6,120) ( IVDH(I), I = 1,6 )

      WRITE(6,180)
      WRITE(6,120) ( IVDN(I), I = 1,2 )

      WRITE(6,185)
      WRITE(6,120) ( IVMF(I), I = 1,4 )

      WRITE(6,190)
      WRITE(6,120) ( IVDR(I), I = 1,3 )

      WRITE(6,200)
      WRITE(6,120) ( IVDSB(I), I = 1,2 )

      WRITE(6,210)
      WRITE(6,120) ( IVQC(I), I = 1,2 )

      WRITE(6,215)
      WRITE(6,120) ( IVMACH(I), I = 1,4 )

      WRITE(6,219)
      WRITE(6,120) ( STDALT(I), I = 1,43 )
101   CONTINUE

         ATMOSPHERIC DATA

         RHO ( ATMOS1 ) = F( ALTD )
         SVEL ( ATMOS2 ) = F( ALTD )
         READ(10,100) ( ATMOS1(I), I = 1,43 )
         READ(10,100) ( ATMOS2(I), I = 1,43 )
         IF ( HCAC .EQ. 0 ) GO TO 103
         WRITE(6,221)
```

```
      WRITE(6,222)
      WRITE(6,223) ( ATMOS1(I), I = 1,43 )

      WRITE(6,224)
      WRITE(6,240) ( ATMOS2(I), I = 1,43 )

  103 CONTINUE
```

LONGITUDINAL DERIVATIVE DATA

LIFT COEFFICIENT

```
CLBAS ( CL1 ) = F( MACH, ALTD, ALFA )
DCLDN ( CL2 ) = F( MACH, ALTD, ALFA )
DCLDF ( CL3 ) = F( MACH, ALTD, ALFA )
DCLDH ( CL4 ) =  F( MACH, DH, ALFA )
FRCLCH ( CL5 ) = F( ALTD, MACH )
DCLDSE ( CL6 ) = F( MACH, DSB, ALFA )
DCLDA ( CL7 ) = F( MACH, DA, ALFA )
FRCLCA ( CL8 ) = F( ALTD, MACH )
CLQ ( CL9 ) = F( MACH, ALTD, ALFA )
CLA ( CL10 ) = F( MACH, ALTD, ALFA )

READ(2,100) ((( CL1(I,J,K), K = 1,22 ), J = 1,4 ), I = 1,4 )
READ(2,100) ((( CL2(I,J,K), K = 1,22 ), J = 1,4 ), I = 1,4 )
READ(2,100) ((( CL3(I,J,K), K = 1,22 ), J = 1,4 ), I = 1,4 )
READ(2,100) ((( CL4(I,J,K), K = 1,22 ), J = 1,6 ), I = 1,4 )
READ(2,100) (( CL5(I,J), J = 1,4 ), I = 1,4 )
READ(2,100) ((( CL6(I,J,K), K = 1,22 ), J = 1,2 ), I = 1,4 )
READ(2,100) ((( CL7(I,J,K), K = 1,22 ), J = 1,5 ), I = 1,4 )
READ(2,100) (( CL8(I,J), J = 1,4 ), I = 1,4 )
READ(2,100) ((( CL9(I,J,K), K = 1,22 ), J = 1,4 ), I = 1,4 )
READ(2,100) ((( CL10(I,J,K), K = 1,22 ), J = 1,4 ), I = 1,4 )
```

```
      IF ( HCCL .EQ. 0 ) GO TO 201

      WRITE(6,225)

      WRITE(6,230)
      DO 232 I = 1,4
        DO 232 J = 1,4
          WRITE(6,236) IVMACH(I), IVALTD(J)
            WRITE(6,240) ( CL1(I,J,K), K = 1,22 )
232   CONTINUE

      WRITE(6,250)
   DO 252 I = 1,4
     DO 252 J = 1,4
       WRITE(6,256) IVMACH(I), IVALTD(J)
       WRITE(6,260) ( CL2(I,J,K), K = 1,22 )
252 CONTINUE

      WRITE(6,270)
   DO 272 I = 1,4
     DO 272 J = 1,4
       WRITE(6,276) IVMACH(I), IVALTD(J)
       WRITE(6,260) ( CL3(I,J,K), K = 1,22 )
272 CONTINUE

      WRITE(6,280)
   DO 282 I = 1,4
     DO 282 J = 1,6
       WRITE(6,286) IVMACH(I), IVDN(J)
       WRITE(6,290) ( CL4(I,J,K), K = 1,22 )
282 CONTINUE

      WRITE(6,300)
   DO 302 I = 1,4
     WRITE(6,306) IVALTD(I)
     WRITE(6,290) (CL5(I,J), J = 1,4 )
302 CONTINUE

      WRITE(6,310)
   DO 312 I = 1,4
     DO 312 J = 1,2
       WRITE(6,316) IVMACH(I), IVDSB(J)
       WRITE(6,290) ( CL6(I,J,K), K = 1,22 )
312 CONTINUE

      WRITE(6,320)
   DO 322 I = 1,4
```

```
      DO 322 J = 1,5
        WRITE(6,324) IVMACH(I), IVDA(J)
        WRITE(6,290) ( CL7(I,J,K), K = 1,22 )
322   CONTINUE

        WRITE(6,330)
      DO 332 I = 1,4
       WRITE(6,334)  IVALTD(I)
       WRITE(6,240)  (CL8(I,J), J =  1,4 )
332   CONTINUE

        WRITE(6,340)
      DO 342 I = 1,4
       DO 342 J = 1,4
        WRITE(6,344) IVMACH(I), IVALTD(J)
        WRITE(6,120) ( CL9(I,J,K), K = 1,22 )
342   CONTINUE

        WRITE(6,350)
      DO 352 I = 1,4
       DO 352 J = 1,4
        WRITE(6,354) IVMACH(I), IVALTD(J)
        WRITE(6,120) ( CL10(I,J,K), K = 1,22 )
352   CONTINUE

201     CONTINUE
```

DRAG COEFFICIENT DATA

PITCHING MOMENT COEFFICIENT DATA

LATERAL-DIRECTIONAL DERIVATIVES

YAWING MOMENT COEFFICIENT DATA

ROLLING MOMENT COEFFICIENT DATA

SIDE FORCE COEFFICIENT DATA

```
      SECTICN THREE:  TEST FLIGHT CONCITICN INFUTS

      THIS SECTION INPUTS THE VALUES CF INCEPENDENT VARIABLES
      DESCRIBING THE AIRCRAFT FLIGHT CONDITION.  A HARDCOPY
      OF THE INPUTS IS GENERATED AUTOMATICALLY WITH EACH PROGRAM
      RUN.

      IF ( TFC .EQ. 0 ) GO TO 1619
      DATA   MACH/.6/, ALTD/40000./, ALFA/20./, BETA/-6./
      DATA   Q/.2/, ALFADT/.4/, P/.5/, R/.5/
1619  CONTINUE

      IF ( TAC .EQ. 0 ) GO TO 1621
      CALL ONEVAR( STDALT, 43, ATMOS1, ALTC, 3, RHO )
      CALL ONEVAR( STDALT, 43, ATMOS2, ALTC, 3, SVEL )
      VT = MACH * SVEL
      QC = .5 * RHO * VT**2
1621  CONTINUE

      IF ( TCSC .EQ. 0 ) GO TO 1623
      DATA   DAL/12.5/, DAR/-12.5/, DNL/25./, DNR/25./,
     #       DFL/20./, DFR/20./, DHL/-6./, DHR/-6./,
     #       DRL/-30./, DRR/-30./, DSB/60./

      DDA = ( DAL - DAR )
      DF = ( DFL + DFR ) / 2
      DDF = ( DFL - DFR )
```

```
      DH = ( DFL + DHR ) / 2
      DN = ( DNL + DNR ) / 2
      DDN = ( DNR - DNL )
      DR = ( DFL + DRR ) / 2
      DT = ( DFL - DHR )
1623  CONTINUE
      IF ( FCFC .EQ. 0 ) GO TO 1625
      WRITE(6,2000)
      WRITE(6,2010)
      WRITE(6,120) ALFA, BETA, ALTD, MACH, QC, Q, ALFADT
      WRITE(6,2015)
      WRITE(6,120) VT, P, R
      WRITE(6,2020)
      WRITE(6,120) DAL, DAR, DDA, DFL, DFR, DF, DDF
      WRITE(6,2022)
      WRITE(6,120) DNL, DNR, DN, DDN, DHL, DHR, DH, DT
      WRITE(6,2024)
      WRITE(6,120) DRL, DRR, DR, DSB
1625  CONTINUE
```

```
      SECTION 4:  AERODYNAMIC BUILD-UP

      THIS SECTION MAKES CALLS TO INTERPOLATION SUBROUTINES WHICH
      INTERPOLATE THE TABULATED DATA FOR THE PROPER VALUES OF THE
      AERODYNAMIC DERIVATIVES FOR THE GIVEN FLIGHT CONDITION.  THE
      DERIVATIVES ARE THEN SUMMED TO CALCULATE THE STATIC AND
      DYNAMIC COEFFICIENTS FROM WHICH THE TOTAL COEFFICIENT IS
      FORMED.  REFER TO THE NOTES IN SECTION ONE FOR VERIFICATION
      OF HARDCOPY OUTPUT.


      LIFT COEFFICIENT


      CLBAS

      CALL THRVAR( IVMACH, IVALTD, IVALF1, 4, 4, 22, CL1, WSMXY,
     #             WSMX, MACH, ALTD, ALFA, 3, 3, 3, CLBAS )

      DCLDN

      CALL THRVAR( IVMACH, IVALTD, IVALF1, 4, 4, 22, CL2, WSMXY,
     #             WSMX, MACH, ALTD, ALFA, 3, 3, 3, DCLDN )

      DCLDF

      CALL THRVAR( IVMACH, IVALTD, IVALF1, 4, 4, 22, CL3, WSMXY,
     #             WSMX, MACH, ALTD, ALFA, 3, 3, 3, DCLDF )

      DCLDHL

      CALL THRVAR( IVMACH, IVDH, IVALF1, 4, 6, 22, CL4, WSMXY,
     #             WSMX, MACH, DHL, ALFA, 3, 3, 3, DCLDHL )

      DCLDHR

      CALL THRVAR( IVMACH, IVDH, IVALF1, 4, 6, 22, CL4, WSMXY,
     #             WSMX, MACH, DHR, ALFA, 3, 3, 3, DCLDHR )

      FRCLDH

      CALL TUVAR( IVALTD, IVMACH, 4, 4, CL5, WSMX, ALTD, MACH,
     #             3, 3, FRCLDH )

      DCLDSB
```

```
      CALL THRVAR( IVMACH, IVDSB, IVALF1, 4, 2, 22, CL6, WSMXY,
     #            WSMX, MACH, DSB, ALFA, 3, 1, 3, DCLDSB )
      DCLDAL
      CALL THRVAR( IVMACH, IVDA, IVALF1, 4, 5, 22, CL7, WSMXY,
     #            WSMX, MACH, DAL, ALFA, 3, 3, 3, DCLDAL )
      DCLDAR
      CALL THRVAR( IVMACH, IVDA, IVALF1, 4, 5, 22, CL7, WSMXY,
     #            WSMX, MACH, DAR, ALFA, 3, 3, 3, DCLDAR )
      FRCLDA
      CALL TUVAR( IVALTD, IVMACH, 4, 4, CL8, WSMX, ALTD, MACH,
     #            3, 3, FRCLDA )
      CLQ
      CALL THRVAR( IVMACH, IVALTD, IVALF1, 4, 4, 22, CL9, WSMXY,
     #            WSMX, MACH, ALTD, ALFA, 3, 3, 3, CLQ )
      CLA
      CALL THRVAR( IVMACH, IVALTD, IVALF1, 4, 4, 22, CL10, WSMXY,
     #            WSMX, MACH, ALTD, ALFA, 3, 3, 3, CLA )
      STATIC LIFT COEFFICIENT
      CLST = CLBAS + ( DCLDN * DN ) + ( DCLDF * DF )
     #          + ( DCLDHL + DCLDHR ) * FRCLDF / 2
     #             + DCLDSB + ( DCLDAL + DCLDAR ) * FRCLDA
      DYNAMIC LIFT COEFFICIENT
      CLDYN = CLQ * ( C * C ) / ( 2 * VT )
     #             + CLA * ( ALFADT * C ) / ( 2 * VT )
      TOTAL LIFT COEFFICIENT
      CL = CLST + CLDYN
    IF ( CLCLT .EQ. 0 ) GO TO 1050
    WRITE(6,1000)
```

```
      WRITE(6,101C)
      WRITE(6,260) CLBAS, DCLDN, DCLDF, DCLDHL, DCLDHR, FRCLDH

      WRITE(6,102C)
      WRITE(6,290) DCLDSB, DCLDAL, DCLDAR, FRCLDA, CLQ, CLA

      WRITE(6,103C)
      WRITE(6,260) CLST, CLDYN, CL

1050  CONTINUE
```

```
DRAG COEFFICIENT

PITCHING MOMENT COEFFICIENT


LATERAL-DIRECTIONAL DERIVATIVES


YAWING MOMENT COEFFICIENT

ROLLING MOMENT COEFFICIENT

SIDE FORCE COEFFICIENT                                        /


WRITE(6,5000)
WRITE(6,240) CD, CL, CM, CN, CR, CY
```

SECTICN 5: OUTPUT AND CONTROL

```
100   FORMAT(8F10.4 )
105   FORMAT('1',//,23X,'INDEPENDENT VARIABLE TABULATED VALUES')
110   FORMAT(///,18X,'REFERENCE ANGLE OF ATTACK VALUES',
     #          ' - LONGITUDINAL DATA',/)
112   FORMAT(///,14X,'REFERENCE ANGLE OF ATTACK VALUES',
     #          ' - LATERAL-DIRECTIONAL DATA',/)
120   FORMAT(/,8F10.1)
130   FORMAT(///,20X,'REFERENCE ALTITUDE VALUES',/)
140   FORMAT(///,20X,'REFERENCE SIDESLIP ANGLE VALUES',/)
150   FORMAT(///,20X,'REFERENCE AILERCN DEFLECTION VALUES',/)
160   FORMAT(///,20X,'REFERENCE T.E. FLAP DEFLECTION VALUES',/)
170   FORMAT(///,20X,'REFERENCE HORIZ. TAIL DEFLECTION VALUES',/)
180   FORMAT(///,20X,'REFERENCE L.E. FLAP DEFLECTION VALUES',/)
185   FORMAT(///,20X,'REFERENCE MANEUVERING FLAP < LEF > VALUES',/)
190   FORMAT(///,20X,'REFERENCE RUDDER DEFLECTION VALUES',/)
200   FORMAT(///,20X,'REFERENCE SPEED BRAKE DEFLECTION VALUES',/)
210   FORMAT(///,20X,'REFERENCE DYNAMIC PRESSURE VALUES',/)
215   FORMAT(///,20X,'REFERENCE MACH NUMBER VALUES',/)
219   FORMAT(///,20X,'ATMOSPHERIC TABLE ALTITUDE VALUES',/)
220   FORMAT(/,8F10.5)
221   FORMAT('1',///,20X,'STANDARD DAY ATMOSPHERIC TABLES',/)
222   FORMAT(///,20X,'STANDARD DAY ATMOSPHERIC DENSITY')
223   FORMAT(/,8F10.7)
```

```
224    FORMAT(///,20X,'STANDARD DAY SONIC VELOCITY')
225    FORMAT('1',///,26X,'LIFT COEFFICIENT DERIVATIVE DATA')
230    FORMAT(////,18X,'LIFT COEFFICIENT - BASIC CONFIGURATION  ',
      #           '< CLBAS >',//)
236    FORMAT(//,10X,'MACH NO. = ',F6.2,5X,'ALTD. = ',F8.2,5X,'ALFA =',
      #           '   ALL',)
240    FORMAT(/,8F10.2)
250    FORMAT(////,18X,'LIFT INCREMENT DUE TO LEF DEFLECTION    ',
      #           '< DCLDN >',//)
256    FORMAT(//,5X,'MACH NO. =',F5.1,5X,'ALTD. =',F7.1,5X,'ALFA = ALL')
260    FORMAT(/,8F10.4)
270    FORMAT(////,18X,'LIFT INCREMENT DUE TO TEF DEFLECTION    ',
      #           '< DCLDF >',//)
276    FORMAT(//,5X,'MACH NO. =',F5.1,5X,'ALTD. =',F7.1,5X,'ALFA = ALL')
280    FORMAT(////,13X,'LIFT INCREMENT DUE TO HORIZONTAL TAIL',
      #           ' DEFLECTION   < DCLDH >',//)
286    FORMAT(//,5X,'MACH NO. =',F5.1,5X,'DN =',F5.1,5X,' ALFA = ALL')
290    FORMAT(/,8F10.3)
300    FORMAT(////,5X,'FLEX/RIGIDITY FACTOR FOR LIFT DUE TO',
      #           ' HORIZONTAL TAIL DEFLECTION  < FRCLDH >',//)
306    FORMAT(//,5X,'ALTD. =',F7.1,5X,'MACH NO. = ALL')
310    FORMAT(////,13X,'LIFT INCREMENT DUE TO SPEED BRAKE',
      #           ' DEFLECTION   < DCLDSB >',//)
316    FORMAT(//,5X,'MACH NO. =',F5.1,5X,'DSB. =',F5.1,5X,'ALFA = ALL')
320    FORMAT(////,17X,'LIFT INCREMENT DUE TO AILERON',
      #           ' DEFLECTION   < DCLDA >',//)
324    FORMAT(//,5X,'MACH NO. =',F5.1,5X,'DA =',F5.1,5X,' ALFA = ALL')
330    FORMAT(////,9X,'FLEX/RIGIDITY FACTOR FOR LIFT DUE TO',
      #           ' AILERON DEFLECTION   < FRCLDA >',//)
```

```
334   FORMAT(//,5X,'ALTD. =',F7.1,5X,'MACH NC. = ALL')

340   FORMAT(////,27X,'LIFT DUE TO PITCH RATE   < CLQ >',//)

344   FORMAT(//,5X,'MACH NO. =',F5.1,5X,'ALTD. =',F7.1,5X,'ALFA = ALL')

350   FORMAT(////,23X,'LIFT DUE TO ANGLE CF ATTACK RATE',
     #           '   < CLA >',//)

1000  FORMAT(////,10X,'OUTPUT VALUES CF LIFT COEFFICIENT',
     #           ' DERIVATIVES',///)

1010  FORMAT(4X,'CLBAS',5X,'DCLDN',5X,'DCLDF',5X,'DCLDHL',4X,'DCLDHR',
     #           4X,'FRCLDH')

1020  FORMAT(///,4X,'DCLDSB',4X,'DCLDAL',4X,'CCLDAR',4X,'FRCLDA',
     #           6X,'CLQ',6X,'CLA')

1030  FORMAT(///,5X,'CLST',6X,'CLDYN',3X,'CL TOTAL')

2000  FORMAT('1',////,23X,'FLIGHT CONDITION PARAMETERS',///)

2010  FORMAT(6X,'ALFA',6X,'BETA',4X,'ALTD',8X,'MACH',3X,'DYNPRESS',7X,
     #           'Q',6X,'ALFADT')

2015  FORMAT(//,7X,'VT',9X,'P',9X,'R')

2020  FORMAT(//,7X,'DAL',7X,'DAR',7X,'DDA',8X,'DFL',6X,'DFR',8X,'DF',
     #           8X,'CDF')

2022  FORMAT(//,7X,'DNL',7X,'DNR',7X,'DN',8X,'DDN',8X,'DHL',8X,'DHR',
     #           7X,'CH',8X,'DT')

2024  FORMAT(//,7X,'DRL',7X,'DRR',7X,'DR',8X,'DSB')

5000  FORMAT('1',/////,10X,'TOTAL AERCDYNAMIC COEFFICIENTS',//,10X,
     #           'CD',10X,'CL',10X,'CM',10X,'CN',10X,'CR',10X,'CY',//)

         SECTION SIX:  SUBROUTINES

      STOP
      END
```

APPENDIX D

SUBROUTINES

```
      THE FOLLOWING SUBROUTINES ARE INCORPORATED IN THE AERODYNAMIC
      BUILDUP IN SECTION SIX.  THEY ARE BASIC INTERPOLATION ROUTINES
      FOR DETERMINING THE VALUES OF FLIGHT DERIVATIVES AT FLIGHT
      CONDITIONS OTHER THAN THOSE FOR WHICH DATA IS TABULATED. A
      BRIEF EXPLANATION IS PROVIDED AT THE BEGINNING OF EACH.


      SUBROUTINE ONEVAR (Z,NZ,FZ,ZIN,NDEGZ,ANS)

  SUBROUTINE ONEVAR INTERPOLATES A FUNCTION OF ONE VARIABLE USING
  LAGRANGIAN POLYNOMIALS OF DEGREE SPECIFIED.
  SPACING OF DATA POINTS NEED NOT BE UNIFORM.
  FUNCTION SHOULD BE SMOOTH IN ALL DIMENSIONS.
  INDEPENDENT VARIABLE MUST BE GIVEN IN INCREASING ORDER.

  VARIABLES:
    Z: ARRAY OF VALUES OF THE INDEPENDENT VARIABLE
   NZ: DIMENSION OF THE ARRAY OF INDEPENDENT VARIABLE
   FZ: ARRAY OF VALUES OF THE FUNCTION EVALUATED AT THE POINTS
          SPECIFIED IN Z. THE DIMENSION OF FZ IS NZ.
   ZIN: INPUT VALUES OF THE INDEPENDENT VARIABLE
              (THE POINT AT WHICH THE FUNCTION IS EVALUATED)
   NDEGZ: THE DEGREE OF THE POLYNOMIAL FITTED TO THE FUNCTION.
          NDEGZ SHOULD NOT BE GREATER THAN (NZ-1).
   ANS: THE INTERPOLATED VALUE OF THE FUNCTION

      DIMENSION FZ(NZ),Z(NZ)

      IF ((NDEGZ+1).GT.NZ) NDEGZ=NZ-1
      DO 10 I=1,NZ
      THIS=Z(I)-ZIN
      IF (THIS.GE.0.) GO TO 20
   10 CONTINUE
      I=NZ
   20 NZLO=I-(INT(FLOAT(NDEGZ)/2.)+1)
      NZHI=NZLO+NDEGZ
   30 IF (NZLO.GE.1) GO TO 40
      NZLO=NZLO+1
      NZHI=NZHI+1
      GO TO 30
   40 IF (NZHI.LE.NZ) GO TO 50
      NZLO=NZLO-1
      NZHI=NZHI-1
      GO TO 40
   50 CONTINUE
```

```
      INITIALIZATION

        ANS=0.0

      COMPUTE INTERPOLATED VALUES

        DO 70 L=NZLO,NZHI
        TERM=FZ(L)
        DO 60 M=NZLO,NZHI
        IF (L.EQ.M) GO TO 60
        TERM=TERM*(ZIN-Z(M))/(Z(L)-Z(M))
 60     CONTINUE
        ANS=ANS+TERM
 70     CONTINUE
        RETURN
        END
```

```
      SUBROUTINE TUVAR (Y,Z,NY,NZ,FYZ,FY,YIN,ZIN,NDEGY,NDEGZ,ANS)

   SUBROUTINE TUVAR INTERPOLATES A FUNCTION OF TWC VARIABLES USING
   LAGRANGIAN PCLYNOMIALS OF DEGREE SPECIFIED BY THE USER.
   SPACING OF DATA POINTS NEED NOT BE UNIFORM. FUNCTION SHOULD
   BE SMOOTH IN ALL DIMENSIONS. INDEPENDENT VARIABLES SHOULD BE
   GIVEN IN INCREASING ORDERS.

   VARIABLES:
     Y,Z: ARRAYS CF VALUES OF THE TWC INDEPENCENT VARIABLES
     NY,NZ: DIMENSIONS OF THE ARRAYS OF INDEPENDENT VARIABLES
     FYZ: ARRAY OF VALUES OF THE FUNCTION EVALUATED AT THE POINTS
            SPECIFIED IN Y & Z. THE DIMENSION CF FYZ = (NY,NZ)
     FY: SUBRCUTINE WORKSPACE OF DIMENSION (NY)
     YIN,ZIN: INPUT VALUES OF THE TWC INDEPENCENT VARIABLES
                   (THE POINT AT WHICH THE FUNCTION IS EVALUATED)
     NDEGY,NDEGZ: THE DESIRED DEGREE OF THE INTERPOLATING POLYNOMIAL IN
                   THE Y & Z DIMENSIONS RESPECTIVELY.
                   NDEGY SHOULD BE LESS THAN CR EQUAL TO NY-1,
                   AND NDEGZ LESS THAN OR EQUAL TO NZ-1.
     ANS: THE INTERPOLATED VALUE OF THE FUNCTION

   METHOD. FIRST, THE PRCGRAM INTERPOLATES FOR THE GIVEN VALUE OF ZIN
   FOR ALL COMBINATIONS OF Y. THESE ARE STORED IN ARRAY FY.
   THESE VALUES ARE THEN INTERPOLATED FOR THE GIVEN VALUE OF YIN
   WHICH YIELCS ANS.

      DIMENSICN FYZ(NY,NZ),Y(NY),Z(NZ),FY(NY)

      FIRST SECTICN SELECTS POINTS FOR INTERPOLATION

      IF ((NCEGZ+1).GT.NZ) NDEGZ=NZ-1
      DO 10 I=1,NZ
      THIS=Z(I)-ZIN
      IF (THIS.GE.0.) GO TO 20
10    CONTINUE
      I=NZ
20    NZLO=I-(INT(FLOAT(NDEGZ)/2.)+1)
      NZHI=NZLO+NCEGZ
30    IF (NZLC.GE.1) GO TO 40
      NZLO=NZLC+1
      NZHI=NZHI+1
      GO TO 30
40    IF (NZHI.LE.NZ) GO TO 50
      NZLO=NZLO-1
      NZHI=NZHI-1
      GO TO 40
```

```
50    CONTINUE
      IF ((NDEGY+1).GT.NY) NDEGY=NY-1
      DO 60 I=1,NY
      THIS=Y(I)-YIN
      IF (THIS.GE.0.) GO TO 70
60    CONTINUE
      I=NY
70    NYLO=I-(INT(FLOAT(NDEGY)/2.)+1)
      NYHI=NYLO+NDEGY
80    IF (NYLO.GE.1) GO TO 90
      NYLO=NYLO+1
      NYHI=NYHI+1
      GO TO 80
90    IF (NYHI.LE.NY) GO TO 100
      NYLO=NYLO-1
      NYHI=NYHI-1
      GO TO 90
100   CONTINUE

    INITIALIZATION

      ANS=0.0
      DO 110 J=NYLO,NYHI
      FY(J)=0.0
110   CONTINUE
   COMPUTE INTERPOLATED VALUES
      DO 140 J=NYLO,NYHI
      DO 130 L=NZLO,NZHI
      TERM=FYZ(J,L)
      DO 120 M=NZLO,NZHI
      IF (L.EQ.M) GO TO 120
      TERM=TERM*(ZIN-Z(M))/(Z(L)-Z(M))
120   CONTINUE
      FY(J)=FY(J)+TERM
130   CONTINUE
140   CONTINUE
      DO 160 L=NYLO,NYHI
      TERM=FY(L)
      DO 150 M=NYLO,NYHI
      IF (L.EQ.M) GO TO 150
      TERM=TERM*(YIN-Y(M))/(Y(L)-Y(M))
150   CONTINUE
      ANS=ANS+TERM
160   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE THRVAR (X,Y,Z,NX,NY,NZ,FXYZ,FXY,FX,XIN,YIN,ZIN,NDEGX,
     1                   NDEGY,NDEGZ,ANS)

   SUBROUTINE THRVAR INTERPOLATES A FUNCTION OF THREE VARIABLES USING
   LAGRANGIAN POLYNOMIALS OF DEGREE SPECIFIED BY THE USER.
   SPACING OF DATA POINTS NEED NOT BE UNIFORM. FUNCTION SHOULD
   BE SMOOTH IN ALL DIMENSIONS. INDEPENDENT VARIABLES MUST BE GIVEN
   IN INCREASING ORDERS.

   VARIABLES:
     X,Y,Z: ARRAYS OF VALUES OF THE THREE INDEPENDENT VARIABLES
     NX,NY,NZ: DIMENSIONS OF THE ARRAYS OF INDEPENDENT VARIABLES
     FXYZ: ARRAY OF VALUES OF THE FUNCTION EVALUATED AT THE POINTS
           SPECIFIED IN X,Y & Z. THE DIMENSION OF FXYZ = (NX,NY,NZ)
     FXY,FX: SUBROUTINE WORKSPACES OF THE APPROPRIATE DIMENSIONS
                 I.E. FXY(NX,NY),FX(NX)
     XIN,YIN,ZIN: INPUT VALUES OF THE THREE INDEPENDENT VARIABLES
                 (THE POINT AT WHICH THE FUNCTION IS EVALUATED)
     NDEGX,NDEGY,NDEGZ: THE DESIRED DEGREE OF THE INTERPOLATING
                 POLYNOMIAL IN THE X, Y & Z DIMENSIONS RESPECTIVELY.
                 NDEGX SHOULD BE LESS THAN OR EQUAL TO NX-1,
                 NDEGY SHOULD BE LESS THAN OR EQUAL TO NY-1,
                 AND NDEGZ LESS THAN OR EQUAL TO NZ-1.
     ANS: THE INTERPOLATED VALUE OF THE FUNCTION
   METHOD. FIRST, THE PROGRAM INTERPOLATES FOR THE GIVEN VALUE OF ZIN
   FOR ALL COMBINATIONS OF X & Y. THESE ARE STORED IN ARRAY FXY.
   SUBSEQUENTLY THE VALUE OF YIN IS INTERPOLATED FOR ALL VALUES OF
   X IN FXY, YIELDING FX. THEN XIN IS INTERPOLATED YIELDING ANS.

      DIMENSION FXYZ(NX,NY,NZ),X(NX),Y(NY),Z(NZ),FXY(NX,NY),FX(NX)

      FIRST SECTION SELECTS POINTS FOR INTERPOLATION

      IF ((NDEGX+1).GT.NX) NDEGX=NX-1
      DO 10 I=1,NX
      THIS=X(I)-XIN
      IF (THIS.GE.0.) GO TO 20
10    CONTINUE
      I=NX
20    NXLO=I-(INT(FLOAT(NDEGX)/2.)+1)
      NXHI=NXLO+NDEGX
30    IF (NXLO.GE.1) GO TO 40
      NXLO=NXLO+1
      NXHI=NXHI+1
      GO TO 30
40    IF (NXHI.LE.NX) GO TO 50
      NXLO=NXLO-1
```

```
      NXHI=NXHI-1
      GO TO 40
50    CONTINUE
      IF ((NDEGY+1).GT.NY) NDEGY=NY-1
      DO 60 I=1,NY
      THIS=Y(I)-YIN
      IF (THIS.GE.0.) GO TO 70
60    CONTINUE
      I=NY
70    NYLO=I-(INT(FLOAT(NDEGY)/2.)+1)
      NYHI=NYLO+NDEGY
80    IF (NYLO.GE.1) GO TO 90
      NYLO=NYLO+1
      NYHI=NYHI+1
      GO TO 80
90    IF (NYHI.LE.NY) GO TO 100
      NYLO=NYLO-1
      NYHI=NYHI-1
      GO TO 90
100   CONTINUE
      IF ((NDEGZ+1).GT.NZ) NDEGZ=NZ-1
      DO 110 I=1,NZ
      THIS=Z(I)-ZIN
      IF (THIS.GE.0.) GO TO 120
110   CONTINUE
      I=NZ
120   NZLO=I-(INT(FLOAT(NDEGZ)/2.)+1)
      NZHI=NZLO+NDEGZ
130   IF (NZLO.GE.1) GO TO 140
      NZLO=NZLO+1
      NZHI=NZHI+1
      GO TO 130
140   IF (NZHI.LE.NZ) GO TO 150
      NZLO=NZLO-1
      NZHI=NZHI-1
      GO TO 140
150   CONTINUE

     INITIALIZATION

      ANS=0.0
      DO 170 I=1,NX
      DO 160 J=1,NY
      FXY(I,J)=0.0
160   CONTINUE
      FX(I)=0.0
170   CONTINUE
    COMPUTE INTERPOLATED VALUES
```

```
      DO 210 I=NXLO,NXHI
      DO 200 J=NYLO,NYHI
      DO 190 L=NZLO,NZHI
      TERM=FXYZ(I,J,L)
      DO 180 M=NZLO,NZHI
      IF (L.EQ.M) GO TO 180
      TERM=TERM*(ZIN-Z(M))/(Z(L)-Z(M))
180   CONTINUE
      FXY(I,J)=FXY(I,J)+TERM
190   CONTINUE
200   CONTINUE
210   CONTINUE
      DO 240 I=NXLO,NXHI
      DO 230 L=NYLO,NYHI
      TERM=FXY(I,L)
      DO 220 M=NYLO,NYHI
      IF (L.EQ.M) GO TO 220
      TERM=TERM*(YIN-Y(M))/(Y(L)-Y(M))
220   CONTINUE
      FX(I)=FX(I)+TERM
230   CONTINUE
240   CONTINUE
      DO 260 L=NXLO,NXHI
      TERM=FX(L)
      DO 250 M=NXLO,NXHI
      IF (L.EQ.M) GO TO 250
      TERM=TERM*(XIN-X(M))/(X(L)-X(M))
250   CONTINUE
      ANS=ANS+TERM
260   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FORVAR (W,X,Y,Z,NW,NX,NY,NZ,FWXYZ,FWXY,FWX,FW,WIN,XIN,
     1                   YIN,ZIN,NDEGW,NDEGX,NDEGY,NDEGZ,ANS)

C     SUBROUTINE FORVAR INTERPOLATES A FUNCTION OF FOUR VARIABLES USING
C     LAGRANGIAN POLYNOMIALS OF DEGREE SPECIFIED BY THE USER.
C     SPACING OF DATA POINTS NEED NOT BE UNIFORM. FUNCTION SHOULD
C     BE SMOOTH IN ALL DIMENSIONS. INDEPENDENT VARIABLES MUST BE
C     GIVEN IN INCREASING ORDERS.

C     VARIABLES:
C       W,X,Y,Z: ARRAYS OF VALUES OF THE FOUR INDEPENDENT VARIABLES
C       NW,NX,NY,NZ: DIMENSIONS OF THE ARRAYS OF INDEPENDENT VARIABLES
C       FWXYZ: ARRAY OF VALUES OF THE FUNCTION EVALUATED AT THE POINTS
C              SPECIFIED IN W,X,Y & Z. THE DIMENSION OF FWXYZ IS THEN
C              (NW,NX,NY,NZ)
C       FWXY,FWX,FW: SUBROUTINE WORKSPACES OF THE APPROPRIATE DIMENSIONS
C                    I.E. FWXY(NW,NX,NY),FWX(NW,NX),FW(NW)
C       WIN,XIN,YIN,ZIN: INPUT VALUES OF THE FOUR INDEPENDENT VARIABLES
C                        (THE POINT AT WHICH THE FUNCTION IS EVALUATED)
C       NDEGW,NDEGX,NDEGY,NDEGZ: THE DESIRED DEGREE OF THE INTERPOLATING
C                    POLYNOMIAL IN THE W, X, Y & Z DIMENSIONS RESPECTIVELY.
C                    NDEGW SHOULD BE LESS THAN OR EQUAL TO NW-1,
C                    NDEGX SHOULD BE LESS THAN OR EQUAL TO NX-1,
C                    NDEGY SHOULD BE LESS THAN OR EQUAL TO NY-1,
C                    AND NDEGZ LESS THAN OR EQUAL TO NZ-1.
C       ANS: THE INTERPOLATED VALUE OF THE FUNCTION
C
C     METHOD. FIRST, THE PROGRAM INTERPOLATES FOR THE GIVEN VALUE OF ZIN
C     FOR ALL COMBINATIONS OF W, X & Y. THESE ARE STORED IN ARRAY FWXY.
C     SUBSEQUENTLY THE VALUE OF YIN IS INTERPOLATED FOR ALL VALUES OF
C     W & X IN FWXY, YIELDING FWX. THEN XIN IS INTERPOLATED (YIELDING FW)
C     AND WIN IN A LIKE MANNER TO YIELD ANS.
      DIMENSION FWXYZ(NW,NX,NY,NZ),W(NW),X(NX),Y(NY),Z(NZ),FWXY(NW,NX,NY
     1),FWX(NW,NX),FW(NW)
C
C      FIRST SECTION SELECTS POINTS FOR INTERPOLATION
C
      IF ((NDEGW+1).GT.NW) NDEGW=NW-1
      DO 10 I=1,NW
      THIS=W(I)-WIN
      IF (THIS.GE.0.) GO TO 20
   10 CONTINUE
      I=NW
   20 NWLO=I-(INT(FLOAT(NDEGW)/2.)+1)
      NWHI=NWLO+NDEGW
   30 IF (NWLO.GE.1) GO TO 40
      NWLO=NWLO+1
```

```
      NWHI=NWHI+1
      GO TO 30
40    IF (NWHI.LE.NW) GO TO 50
      NWLO=NWLO-1
      NWHI=NWHI-1
      GO TO 40
50    CONTINUE
      IF ((NDEGX+1).GT.NX) NDEGX=NX-1
      DO 60 I=1,NX
      THIS=X(I)-XIN
      IF (THIS.GE.0.) GO TO 70
60    CONTINUE
      I=NX
70    NXLO=I-(INT(FLOAT(NDEGX)/2.)+1)
      NXHI=NXLO+NDEGX
80    IF (NXLO.GE.1) GO TO 90
      NXLO=NXLO+1
      NXHI=NXHI+1
      GO TO 80
90    IF (NXHI.LE.NX) GO TO 100
      NXLO=NXLO-1
      NXHI=NXHI-1
      GO TO 90
100   CONTINUE
      IF ((NDEGY+1).GT.NY) NDEGY=NY-1
      DO 110 I=1,NY
      THIS=Y(I)-YIN
      IF (THIS.GE.0.) GO TO 120
110   CONTINUE
      I=NY
120   NYLO=I-(INT(FLOAT(NDEGY)/2.)+1)
      NYHI=NYLO+NDEGY
130   IF (NYLO.GE.1) GO TO 140
      NYLO=NYLO+1
      NYHI=NYHI+1
      GO TO 130
140   IF (NYHI.LE.NY) GO TO 150
      NYLO=NYLO-1
      NYHI=NYHI-1
      GO TO 140
150   CONTINUE
      IF ((NDEGZ+1).GT.NZ) NDEGZ=NZ-1
      DO 160 I=1,NZ
      THIS=Z(I)-ZIN
      IF (THIS.GE.0.) GO TO 170
160   CONTINUE
      I=NZ
170   NZLO=I-(INT(FLOAT(NDEGZ)/2.)+1)
```

```
      NZHI=NZLO+NDEGZ
180   IF (NZLO.GE.1) GO TO 190
      NZLO=NZLO+1
      NZHI=NZHI+1
      GO TO 180
190   IF (NZHI.LE.NZ) GO TO 200
      NZLO=NZLO-1
      NZHI=NZHI-1
      GO TO 190
200   CONTINUE

    INITIALIZATION

      ANS=0.0
      DO 230 I=NWLO,NWHI
      DO 220 J=NXLO,NXHI
      DO 210 K=NYLO,NYHI
      FWXY(I,J,K)=0.0
210   CONTINUE
      FWX(I,J)=0.0
220   CONTINUE
      FW(I)=0.0
230   CONTINUE
   COMPUTE INTERPOLATED VALUES
      DO 280 I=NWLO,NWHI
      DO 270 J=NXLO,NXHI
      DO 260 K=NYLO,NYHI
      DO 250 L=NZLO,NZHI
      TERM=FWXYZ(I,J,K,L)
      DO 240 M=NZLO,NZHI
      IF (L.EQ.M) GO TO 240
      TERM=TERM*(ZIN-Z(M))/(Z(L)-Z(M))
240   CONTINUE
      FWXY(I,J,K)=FWXY(I,J,K)+TERM
250   CONTINUE
260   CONTINUE
270   CONTINUE
280   CONTINUE
      DO 320 I=NWLO,NWHI
      DO 310 J=NXLO,NXHI
      DO 300 L=NYLO,NYHI
      TERM=FWXY(I,J,L)
      DO 290 M=NYLO,NYHI
      IF (L.EQ.M) GO TO 290
      TERM=TERM*(YIN-Y(M))/(Y(L)-Y(M))
290   CONTINUE
      FWX(I,J)=FWX(I,J)+TERM
300   CONTINUE
```

```
 310   CONTINUE                                          A1420
 320   CONTINUE                                          A1430
       DO 350 I=NWLO,NWHI                                A1440
       DO 340 L=NXLO,NXHI                                A1450
       TERM=FWX(I,L)                                     A1460
       DO 330 M=NXLO,NXHI                                A1470
       IF (L.EQ.M) GO TO 330                             A1480
       TERM=TERM*(XIN-X(M))/(X(L)-X(M))                  A1490
 330   CONTINUE                                          A1500
       FW(I)=FW(I)+TERM                                  A1510
 340   CONTINUE                                          A1520
 350   CONTINUE                                          A1530
       DO 370 L=NWLO,NWHI                                A1540
       TERM=FW(L)                                        A1550
       DO 360 M=NWLO,NWHI                                A1560
       IF (L.EQ.M) GO TO 360                             A1570
       TERM=TERM*(WIN-W(M))/(W(L)-W(M))                  A1580
 360   CONTINUE                                          A1590
       ANS=ANS+TERM                                      A1600
 370   CONTINUE                                          A1610
       RETURN                                            A1620
       END                                               A1630-
```

APPENDIX E

SAMPLE OUTPUT

```
                    LIFT COEFFICIENT DERIVATIVE DATA

             LIFT COEFFICIENT - BASIC CONFIGURATION  < CLBAS >

     MACH NO. =   0.20     ALTD. =      0.0     ALFA =   ALL
-0.35     -0.04      0.26      0.56      0.86      1.10      1.33      1.50
 1.60      1.68      1.70      1.90      1.76      1.60      1.46      1.27
 1.10      0.90      0.70      0.46      0.28      0.10

     MACH NO. =   0.20     ALTD. =  20000.0     ALFA =   ALL
-0.35     -0.04      0.26      0.56      0.86      1.10      1.33      1.50
 1.60      1.68      1.70      1.90      1.76      1.60      1.46      1.27
 1.10      0.90      0.70      0.46      0.28      0.10

     MACH NO. =   0.20     ALTD. =  40000.0     ALFA =   ALL
-0.35     -0.04      0.26      0.56      0.86      1.10      1.33      1.50
 1.60      1.68      1.70      1.90      1.76      1.60      1.46      1.27
 1.10      0.90      0.70      0.46      0.28      0.10

     MACH NO. =   0.20     ALTD. =  60000.0     ALFA =   ALL
-0.35     -0.04      0.26      0.56      0.86      1.10      1.33      1.50
 1.60      1.68      1.70      1.90      1.76      1.60      1.46      1.27
 1.10      0.90      0.70      0.46      0.28      0.10
```

```
        MACH NO. =    0.60      ALTD. =       0.0      ALFA =   ALL
-0.42        -0.05        0.30        0.65        0.94        1.14        1.33        1.48
 1.62         1.67        1.67        1.90        1.76        1.60        1.46        1.27
 1.10         0.90        0.70        0.46        0.28        0.10

        MACH NO. =    0.60      ALTD. = 20000.0      ALFA =   ALL
-0.42        -0.05        0.30        0.65        0.94        1.14        1.35        1.50
 1.63         1.69        1.68        1.90        1.76        1.60        1.46        1.27
 1.10         0.90        0.70        0.46        0.28        0.10

        MACH NO. =    0.60      ALTD. = 40000.0      ALFA =   ALL
-0.42        -0.05        0.30        0.65        0.94        1.14        1.36        1.51
 1.65         1.70        1.70        1.90        1.76        1.60        1.46        1.27
 1.10         0.90        0.70        0.46        0.28        0.10

        MACH NO. =    0.60      ALTD. = 60000.0      ALFA =   ALL
-0.42        -0.05        0.30        0.65        0.94        1.14        1.36        1.51
 1.65         1.70        1.70        1.90        1.76        1.60        1.46        1.27
 1.10         0.90        0.70        0.46        0.28        0.10

        MACH NO. =    0.80      ALTD. =       0.0      ALFA =   ALL
-0.47        -0.06        0.35        0.72        0.95        1.10        1.27        1.40
 1.52         1.61        1.65        1.88        1.75        1.60        1.46        1.27
 1.10         0.90        0.70        0.46        0.28        0.10

        MACH NO. =    0.80      ALTD. = 20000.0      ALFA =   ALL
-0.47        -0.06        0.35        0.72        0.97        1.13        1.30        1.44
```

```
  1.56       1.65       1.70       1.88       1.75       1.60       1.46       1.27
  1.10       0.90       0.70       0.46       0.28       0.10

     MACH NO. =     0.80      ALTD. = 40000.0       ALFA =    ALL
 -0.47      -0.06       0.35       0.72       0.97       1.13       1.32       1.47
  1.58       1.67       1.72       1.88       1.75       1.60       1.46       1.27
  1.10       0.90       0.70       0.46       0.28       0.10

     MACH NO. =     0.80      ALTD. = 60000.0       ALFA =    ALL
 -0.47      -0.06       0.35       0.72       0.97       1.13       1.32       1.47
  1.58       1.67       1.72       1.88       1.75       1.60       1.46       1.27
  1.10       0.90       0.70       0.46       0.28       0.10

     MACH NO. =     0.90      ALTD. =       0.0       ALFA =    ALL
 -0.54      -0.10       0.40       0.80       1.02       1.20       1.35       1.40
  1.56       1.66       1.69       1.88       1.75       1.60       1.46       1.27
  1.10       0.90       0.70       0.46       0.28       0.10

     MACH NO. =     0.90      ALTD. = 20000.0       ALFA =    ALL
 -0.54      -0.10       0.40       0.80       1.04       1.23       1.39       1.45
  1.62       1.71       1.75       1.88       1.75       1.60       1.46       1.27
  1.10       0.90       0.70       0.46       0.28       0.10

     MACH NO. =     0.90      ALTD. = 40000.0       ALFA =    ALL
 -0.54      -0.10       0.40       0.80       1.05       1.25       1.42       1.48
  1.64       1.74       1.76       1.88       1.75       1.60       1.46       1.27
```

1.10	0.90	0.70	0.46	0.28	0.10		

MACH NO. = 0.90 ALTD. = 60000.0 ALFA = ALL

-0.54	-0.10	0.40	0.80	1.05	1.25	1.42	1.48
1.64	1.75	1.78	1.88	1.75	1.60	1.46	1.27
1.10	0.90	0.70	0.46	0.28	0.10		

LIFT INCREMENT DUE TO LEF DEFLECTION < DCLDN >

MACH NO. = 0.2 ALTD. = 0.0 ALFA = ALL

-0.0017	-0.0017	-0.0016	-0.0016	-0.0013	0.0005	0.0010	0.0009
0.0024	0.0044	0.0062	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = 0.2 ALTD. = 20000.0 ALFA = ALL

-0.0017	-0.0017	-0.0016	-0.0016	-0.0013	0.0005	0.0010	0.0009
0.0024	0.0044	0.0062	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = 0.2 ALTD. = 40000.0 ALFA = ALL

-0.0017	-0.0017	-0.0016	-0.0016	-0.0013	0.0005	0.0010	0.0009
0.0024	0.0044	0.0062	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = 0.2 ALTD. = 60000.0 ALFA = ALL

-0.0017	-0.0C17	-0.0016	-0.0016	-0.0013	0.0005	0.0010	0.CC09
0.0024	0.0C44	0.0062	0.0	0.0	0.0	0.0	0.0
C.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = C.6 ALTD. = 0.0 ALFA = ALL

-0.0030	-C.0C28	-0.0023	-0.0012	0.0007	0.0020	0.0025	0.CC27
0.0028	C.0C47	0.0062	0.0	0.0	0.0	0.0	0.C
C.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = C.6 ALTD. = 20000.0 ALFA = ALL

-C.0030	-C.0C28	-0.0023	-0.0012	0.0007	0.0020	0.0025	0.0027
0.0028	0.0C47	0.0062	0.0	0.0	0.0	0.0	0.C
0.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = C.6 ALTD. = 40000.0 ALFA = ALL

-0.0030	-0.0C28	-0.0023	-0.0012	0.0007	0.0020	0.0025	0.C027
C.0028	0.0C47	0.0062	0.0	0.0	0.0	0.0	0.0
C.0	C.0	0.0	0.0	0.0	0.0		

MACH NO. = C.6 ALTD. = 60000.0 ALFA = ALL

-C.0030	-0.0C28	-0.0023	-0.0012	0.0007	0.0020	0.0025	0.C027
0.0028	0.0C47	0.0062	0.0	0.0	0.0	0.0	0.0
C.0	C.0	0.0	0.0	0.0	0.0		

MACH NO. = C.8 ALTD. = 0.0 ALFA = ALL

-0.0032	-C.0C25	-0.0022	-0.0022	-0.0012	0.0003	0.0011	0.C010
0.0004	C.0C04	0.0017	0.0	0.0	0.0	0.0	0.C

```
0.0        0.0        0.0        0.0        0.0        0.0

 MACH NO. =   C.8       ALTD. = 20000.0      ALFA = ALL
-C.0031    -0.0C24    -0.0021    -0.0020    -0.0009     0.0010     0.0023     0.C025
 0.0022     C.0C22     0.0035     0.0        0.0        0.0        0.0        0.C
 0.0        0.0        0.0        0.0        0.0        0.0

 MACH NO. =   C.8       ALTD. = 40000.0      ALFA = ALL
-0.0030    -C.0C23    -0.0020    -0.0019    -0.0008     0.0014     0.0029     0.C032
 C.0031     0.0C32     0.0045     0.0        0.0        0.0        0.0        0.0
 0.0        C.0        0.0        0.0        0.0        0.0

 MACH NO. =   C.8       ALTD. = 60000.0      ALFA = ALL
-0.0030    -0.0C23    -0.0020    -0.0019    -0.0008     0.0015     0.0032     0.C036
 0.0034     0.0C36     0.0048     0.0        0.0        0.0        0.0        0.0
 C.0        C.0        0.0        0.0        C.0        0.0

 MACH NO. =   C.9       ALTD. =      0.0      ALFA = ALL
 0.0025    -C.0C04    -0.0051    -0.0067    -0.0048    -0.0015    -0.0002     0.CC07
 0.0002    -C.0C05    -0.0011     0.0        0.0        0.0        0.0        0.C
 0.0        C.0        0.0        0.0        0.0        0.0

 MACH NO. =   C.9       ALTD. = 20000.0      ALFA = ALL
 0.0027     C.0       -0.0050    -0.0064    -0.0042    -0.0003     0.0015     0.C029
 C.0028     C.0C20     0.0014     0.0        0.0        0.0        0.0        0.C
 0.0        C.0        0.0        0.0        0.0        0.0
```

MACH NO. = C.9 ALTD. = 40000.0 ALFA = ALL

0.0028	0.0	-0.0050	-0.0062	-0.0035	0.0002	0.0025	0.C040
0.0040	0.0C33	0.0C27	0.0	0.0	0.0	0.0	0.C
C.0	0.0	0.0	0.0	0.0	0.0		

MACH NO. = C.9 ALTD. = 60000.0 ALFA = ALL

0.0029	0.0	-0.0050	-0.0062	-0.0035	0.0003	0.0028	0.C045
0.0045	C.0C38	0.0032	0.0	0.0	0.0	0.0	0.0
C.0	C.0	0.0	0.0	0.0	0.0		

FLEX/RIGIDITY FACTOR FOR LIFT DUE TC HORIZONTAL TAIL DEFLECTION < FRCLDH >

ALTD. = C.0 MACH NO. = ALL

C.990	C.920	C.860	0.775

ALTD. = 20C0C.0 MACH NO. = ALL

C.990	C.960	0.928	0.880

ALTD. = 40C0C.0 MACH NO. = ALL

0.995	0.986	0.965	0.945

ALTD. = 6000C.0 MACH NO. = ALL

1.100	0.995	0.985	0.975

FLIGHT CONDITION PARAMETERS

ALFA	BETA	ALTD	MACH	DYNPRESS	Q	ALFADT	
20.0	-6.0	40000.0	0.6	98.8	0.2	0.4	
VT	P	R					
581.1	0.5	0.5					
DAL	DAR	DDA	DFL	DFR	DF	DDF	
12.5	-12.5	25.0	20.0	20.0	20.0	0.0	
DNL	DNR	DN	DDN	DHL	DHR	DH	DT
25.0	25.0	25.0	0.0	-6.0	-6.0	-6.0	0.0
DRL	DRR	DR	DSB				
-30.0	-30.0	-30.0	60.0				

OUTPUT VALUES OF LIFT COEFFICIENT DERIVATIVES

CLBAS	DCLDN	DCLDF	DCLDHL	DCLDHR	FRCLDH
1.3600	0.0025	0.0110	-0.0850	-0.0850	0.9860

DCLDSB	DCLDAL	DCLDAR	FRCLDA	CLQ	CLA
-0.032	0.026	-0.014	1.180	3.300	2.400

CLST	CLDYN	CL TOTAL
1.5408	0.0161	1.5569

LIST OF REFERENCES

1. Naval Safety Center Safety Survey, All Uncontrolled Flight CY76 Thru 10 Jun 82, 1 August 1983.

2. McDonnell Douglas Corporation Report, MDC A7248, F/A-18 Flight Control Description Report, 31 August 1981.

3. McDonnell Douglas Corporation Report, MDC A4379, F/A-18 Flight Control System Schematic Arrangment Drawings, Rev. C, 30 September 1979.

4. McDonnell Douglas Corporation Report, MDC A7813, Volume I, F/A-18 Flight Control System Design Report, 23 December 1982.

5. McDonnell Douglas Corporation Report, MDC A7247, Volumes I and II, F/A-18 Stability and Control Data Report, Rev. B, 15 November 1982.

6. McDonnell Douglas Corporation Report, MDC A7249, F/A-18 Flying Qualities Report, 31 August 1981.

7. Duke, E.L.; Jones, F.P.; Roncoli, R.B.; Development of A Flight Test Maneuver Autopilot for A Highly Maneuverable Aircraft, NASA Ames Research Center, Edwards, CA. Paper presented at the AIAA 21st Aerospace Sciences Meeting, Reno, Nevada, 10-13 January 1983.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943	2
3.	Chairman Department of Aeronautics, Code 67 Naval Postgraduate School Monterey, CA 93943	1
4.	Professor Marle D. Hewett Department of Aeronautics, Code 67-Hj Naval Postgraduate School Monterey, CA 93943	5
5.	LT A. L. Raithel III 10487 Hermanos Road San Diego, CA 92124	5
6.	Naval Air Development Center Code 6012: Mr. Al Piranian Warminster, PA 18974	1
7.	Mr. Lee Duke Mailstop OFDC NASA Dryden Flight Research Facility P.O. Box 273 Edwards, CA 93523	1
8.	Northrop Corporation 3831/82 Flight Controls Research 1 Northrop Ave. Hawthorne, CA 90250	1
9.	Naval Air Systems Command Air 53011B2 ATTN: Mr. Jonah Ottensoser Mr. Tom Lawrence Washington, D.C. 20361	2

	No. Copies
10. Mr. Albert L. Raithel, Jr. VEDA, Inc. 1755 South Jefferson Davis Highway Suite 700 Arlington, VA 22202	2

Thesis 207575
R1473 Raithel
c.1 Development of a flight simulation concept and aerodynamic buildup for investigation of departure prevention systems in tactical aircraft.